MANUAL DE INSTRUÇÕES
OPERAÇÃO E MANUTENÇÃO

DC303

E/S REMOTAS PROFIBUS - PA

ABR / 15

DC303

VERSÃO 3

INTRODUÇÃO

Os equipamentos de campo disponíveis no sistema Profibus-PA não permitiam um sistema híbrido que aceitasse os sinais Profibus e convencionais de Entrada/Saída. Um sistema misto é necessário durante a transição para a tecnologia Profibus. O DC303 permite uma integração fácil entre o Profibus e as Entradas/Saídas Convencionais.

Os equipamentos discretos como chaves de pressão, botoeiras, válvulas On/Off, bombas e esteiras podem ser integradas ao sistema PROFIBUS-PA, usando o DC303. A alimentação e as E/S integradas em um mesmo equipamento, tornam o DC303 um equipamento compacto de fácil uso requerendo menos espaço em painéis quando comparados com outras soluções. O DC303 é parte integrante do SYSTEM302 e pode ser facilmente integrado em outros sistemas suportando o Profibus.

O DC303 permite que as entradas e saídas discretas e analógicas convencionais possam estar disponíveis e facilitar a configuração das estratégias de controle. Usa o conceito de Blocos Funcionais Profibus-PA e torna o sistema homogêneo de modo a fazer com que estes equipamentos possam parecer como simples equipamentos em um barramento Profibus.

O DC303 é um equipamento de baixo custo que pode ser montado em trilho DIN, e que inclui a alimentação e os sinais de barramento Profibus-PA, tudo em uma mesma unidade, tornando-o compacto e requerendo menos espaço em painéis, ao contrário de outras soluções.

Os blocos funcionais de Entrada e Saídas Discretas fornecem grande flexibilidade em estratégias de controle. As E/S discretas convencionais trabalham junto com os equipamentos puramente Profibus integrados numa mesma rede e numa mesma malha de controle. O DC303 é totalmente configurado pelo SYSTEM302 ou por qualquer outra ferramenta de configuração Profibus baseada em EDDL e FDT/DTM. O DC303 pode ser instalado perto de sensores e atuadores, eliminando a necessidade de extensos cabeamentos associados aos painéis e bandejas para as E/S convencionais, com redução do custo do sistema. O DC303 torna possível que as conexões das E/S sejam distribuídas em várias localizações, e é ideal para conectar centros de controle de motores, equipamentos de velocidade variável, atuadores elétricos e válvulas operadas a motores pelo barramento Profibus-PA. Através de seu bloco transdutor, possui um bloco funcional flexível built-in que permite a execução de lógicas.

Obtenha melhores resultados com o DC303 lendo cuidadosamente estas instruções.

NOTA
Nos casos em que o Simatic PDM seja usado como ferramenta de configuração e parametrização, a Smar recomenda que não se faça o uso da opção “Download to Device”. Esta função pode configurar inadequadamente o equipamento. A Smar recomenda que o usuário faça uso da opção “Download to PG/PC” e depois faça uso do Menu Device, onde se tem os menus dos blocos transdutores, funcionais e display e que se atue pontualmente, de acordo com menus e métodos de leitura e escrita.

ATENÇÃO
Este manual é compatível com a versão 3.XX, onde 3 denota a versão de software e XX indica o "release". Portanto, este manual é compatível com todos os "releases" da versão 3.

Exclusão de responsabilidade

O conteúdo deste manual está de acordo com o hardware e software utilizados na versão atual do equipamento. Eventualmente podem ocorrer divergências entre este manual e o equipamento. As informações deste documento são revistas periodicamente e as correções necessárias ou identificadas serão incluídas nas edições seguintes. Agradecemos sugestões de melhorias.

Advertência

Para manter a objetividade e clareza, este manual não contém todas as informações detalhadas sobre o produto e, além disso, ele não cobre todos os casos possíveis de montagem, operação ou manutenção.

Antes de instalar e utilizar o equipamento, é necessário verificar se o modelo do equipamento adquirido realmente cumpre os requisitos técnicos e de segurança de acordo com a aplicação. Esta verificação é responsabilidade do usuário.

Se desejar mais informações ou se surgirem problemas específicos que não foram detalhados e ou tratados neste manual, o usuário deve obter as informações necessárias do fabricante Smar. Além disso, o usuário está ciente que o conteúdo do manual não altera, de forma alguma, acordo, confirmação ou relação judicial do passado ou do presente e nem faz parte dos mesmos.

Todas as obrigações da Smar são resultantes do respectivo contrato de compra firmado entre as partes, o qual contém o termo de garantia completo e de validade única. As cláusulas contratuais relativas à garantia não são nem limitadas nem ampliadas em razão das informações técnicas apresentadas no manual.

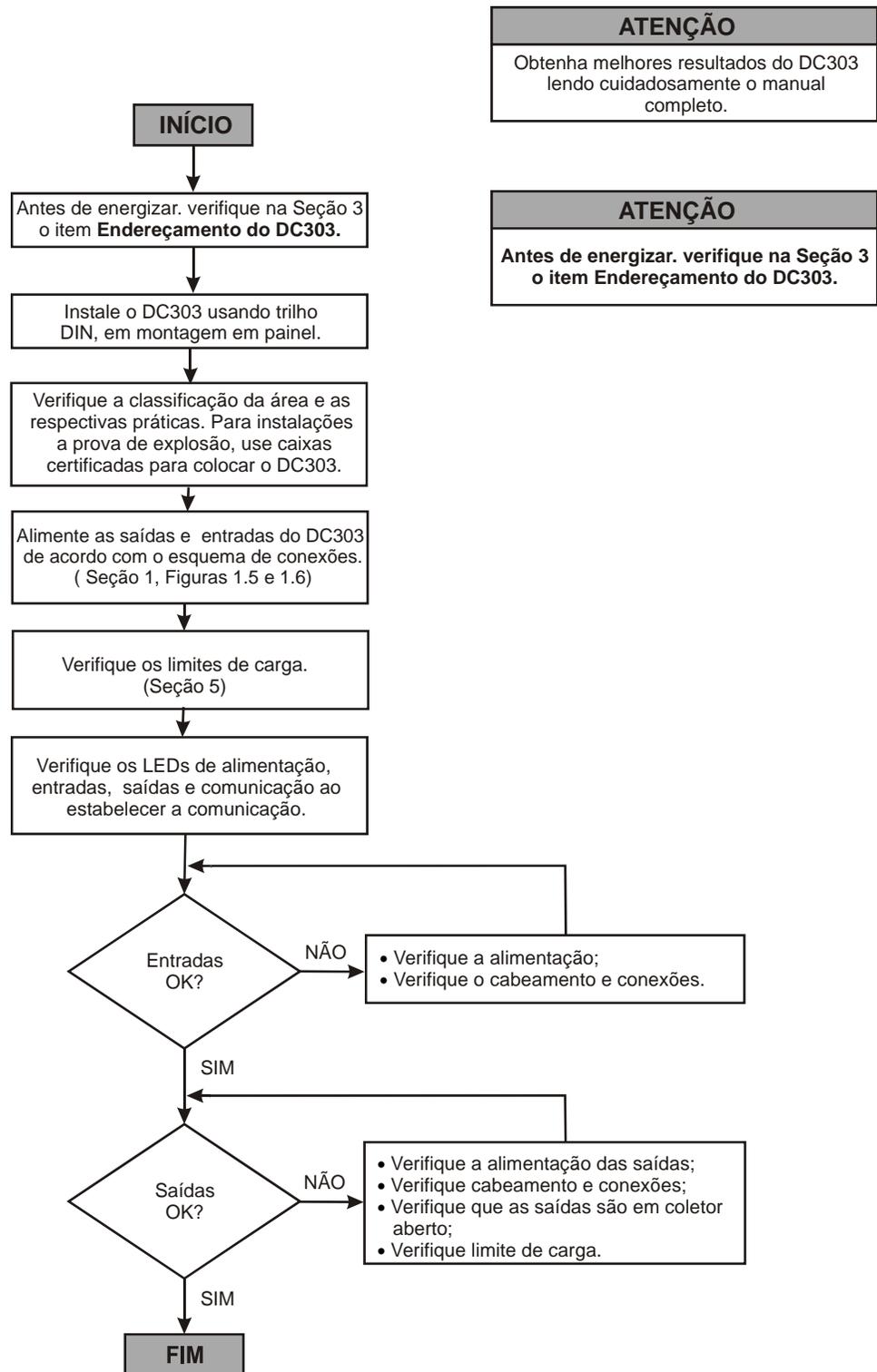
Só é permitida a participação de pessoal qualificado para as atividades de montagem, conexão elétrica, colocação em funcionamento e manutenção do equipamento. Entende-se por pessoal qualificado os profissionais familiarizados com a montagem, conexão elétrica, colocação em funcionamento e operação do equipamento ou outro aparelho similar e que dispõem das qualificações necessárias para suas atividades. A Smar possui treinamentos específicos para formação e qualificação de tais profissionais. Adicionalmente, devem ser obedecidos os procedimentos de segurança apropriados para a montagem e operação de instalações elétricas de acordo com as normas de cada país em questão, assim como os decretos e diretivas sobre áreas classificadas, como segurança intrínseca, prova de explosão, segurança aumentada, sistemas instrumentados de segurança entre outros.

O usuário é responsável pelo manuseio incorreto e/ou inadequado de equipamentos operados com pressão pneumática ou hidráulica, ou ainda submetidos a produtos corrosivos, agressivos ou combustíveis, uma vez que sua utilização pode causar ferimentos corporais graves e/ou danos materiais.

O equipamento de campo que é referido neste manual, quando adquirido com certificado para áreas classificadas ou perigosas, perde sua certificação quando tem suas partes trocadas ou intercambiadas sem passar por testes funcionais e de aprovação pela Smar ou assistências técnicas autorizadas da Smar, que são as entidades jurídicas competentes para atestar que o equipamento como um todo, atende as normas e diretivas aplicáveis. O mesmo acontece ao se converter um equipamento de um protocolo de comunicação para outro. Neste caso, é necessário o envio do equipamento para a Smar ou à sua assistência autorizada. Além disso, os certificados são distintos e é responsabilidade do usuário sua correta utilização.

Respeite sempre as instruções fornecidas neste Manual. A Smar não se responsabiliza por quaisquer perdas e/ou danos resultantes da utilização inadequada de seus equipamentos. É responsabilidade do usuário conhecer as normas aplicáveis e práticas seguras em seu país.

ÍNDICE

Fluxograma de Instalação

* Maiores informações encontram-se na Seção 1 do manual de instalação, configuração e manutenção do **DC303**.
Consulte também o manual de blocos funcionais e manual geral do Profibus.

Seção 1

INSTALAÇÃO

Geral

ATENÇÃO
Antes de energizar, verifique o item **Endereçamento do DC303** na **Seção 3**, neste manual.

A precisão de uma medição em controle depende de várias variáveis. Embora o **DC303** tenha um alto desempenho, uma instalação adequada é necessária para se aproveitar ao máximo os benefícios oferecidos.

De todos os fatores que podem afetar a precisão, as condições ambientais são as mais difíceis de controlar. Entretanto, há maneiras de se reduzir os efeitos de temperatura, umidade e vibração.

Localizando o **DC303** em áreas protegidas de mudanças bruscas ambientais, pode-se melhorar sua performance.

Em ambientes quentes, o **DC303** deve ser instalado de forma a se evitar ao máximo a exposição a raios solares. Deve-se evitar a instalação próxima a linhas ou vasos com altas temperaturas.

A umidade é fatal aos circuitos eletrônicos. Em áreas com alto índice de umidade relativa deve-se certificar da correta instalação e proteção.

Para detalhes de montagem refira às Figuras 1.1 e 1.2.

Montagem

Use o trilho DIN (TS35-DIN EN 50022 ou TS32-DIN EN50035 ou TS15 DIN EN50045), como é mostrado na Figura 1.1 – Montagem Mecânica. O **DC303** pode ser opcionalmente fornecido em uma caixa de distribuição a prova de explosão.

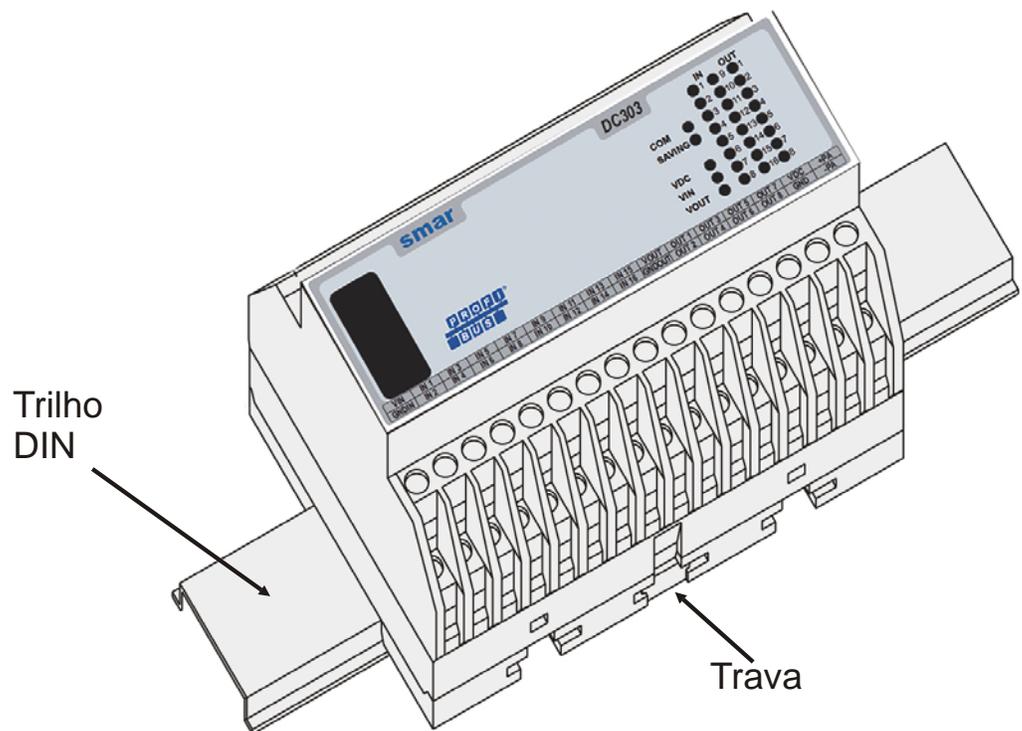
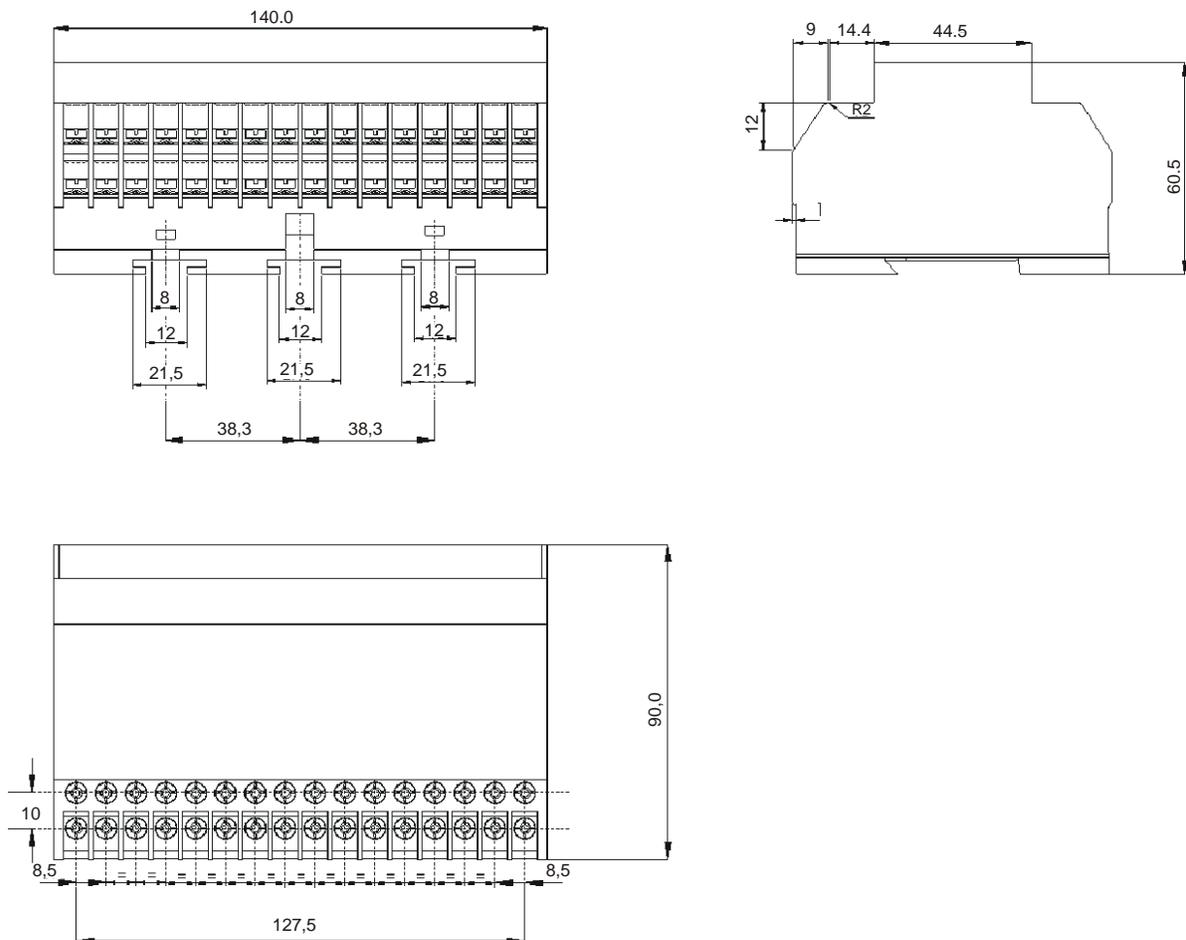



Figura 1.1- Montagem Mecânica

NOTA
As medidas são em mm.

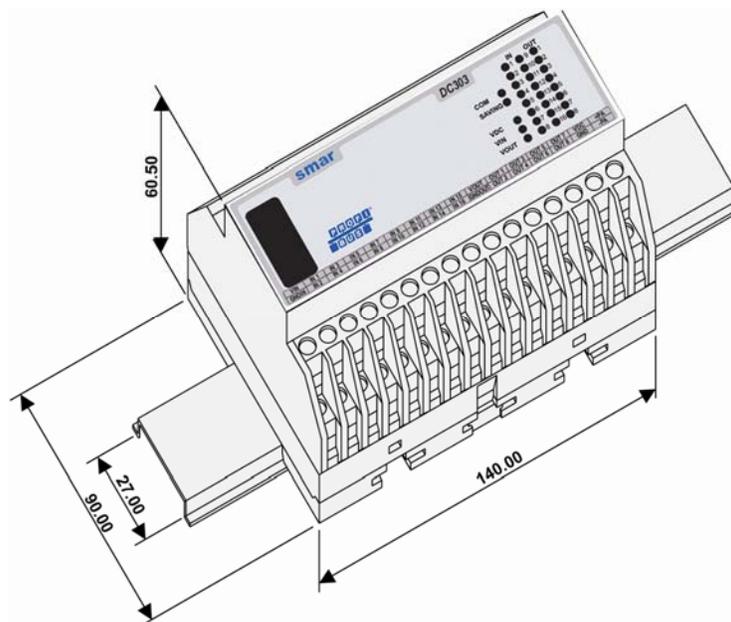



Figura 1.2 – Montagem Mecânica e desenho dimensional do DC303

Conexão Elétrica

Acesse o conjunto de conexão na vista frontal através da etiqueta de conexão para as entradas, saídas e fonte de alimentação. As conexões são feitas através dos parafusos.

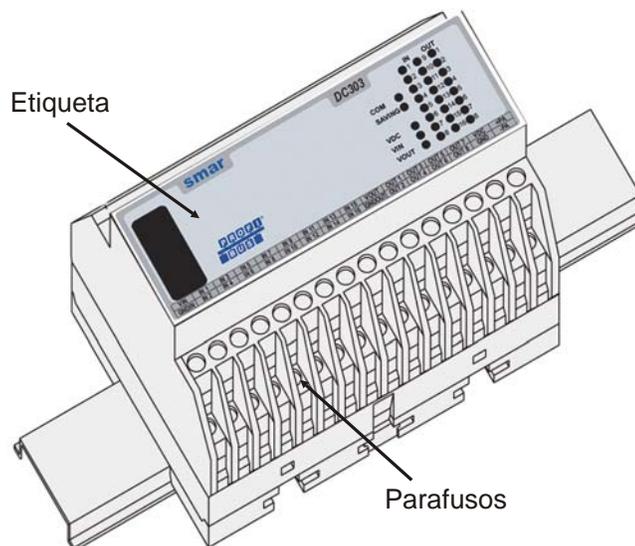



Figura 1.3 – Bloco de conexões

A tabela a seguir descreve os terminais do **DC303**:

Terminal	Superiot (S)	Inferior (I)	Comentário
1	VIN (1S)	GNDIN (1I)	Alimentação auxiliar para as entradas.
2	IN1 (2S)	IN2 (2I)	Entradas digitais.
3	IN3 (3S)	IN4 (3I)	
4	IN5 (4S)	IN6 (4I)	
5	IN7 (5S)	IN8 (5I)	
6	IN9 (6S)	IN10 (6I)	
7	IN11 (7S)	IN12 (7I)	
8	IN13 (8S)	IN14 (8I)	
9	IN15 (9S)	IN16 (9I)	
10	VOUT (10S)	GNDOUT (10I)	Alimentação auxiliar para as saídas.
11	OUT1 (11S)	OUT2 (11I)	Saídas digitais.
12	OUT3 (12S)	OUT4 (12I)	
13	OUT5 (13S)	OUT6 (13I)	
14	OUT7 (14S)	OUT8 (14I)	
15	VDC (15S)	GND (15I)	Alimentação auxiliar para a comunicação.
16	+PA(16S)	-PA (16I)	Sinal de comunicação Profibus PA.

Tabela1.1 – Bloco de conexões

As conexões utilizadas devem ser feitas de acordo com a aplicação. Por exemplo, refira-se as figuras 1.4 e 1.5.

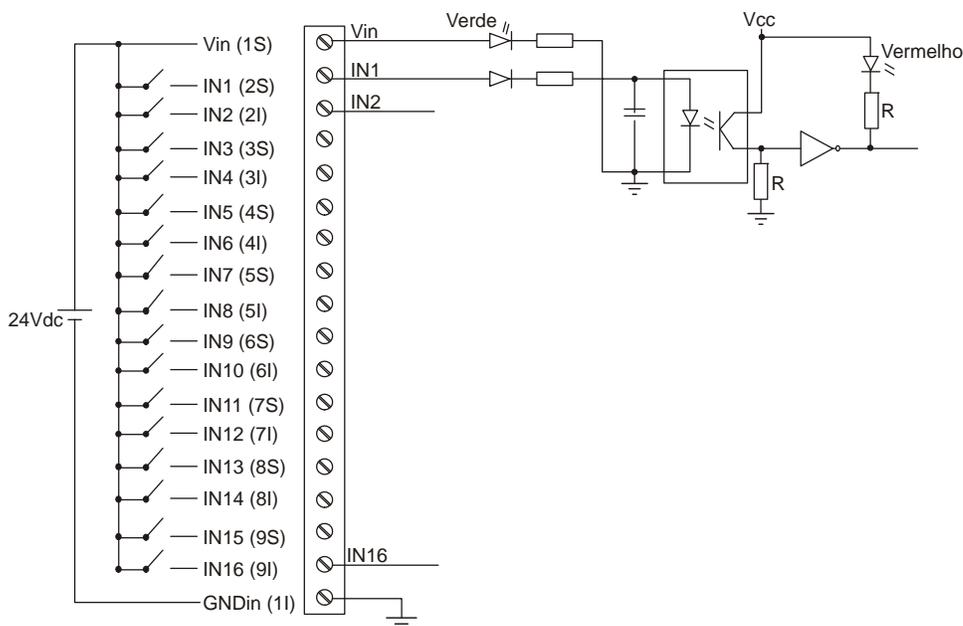



Figura 1.4 – Exemplo de conexões de entrada

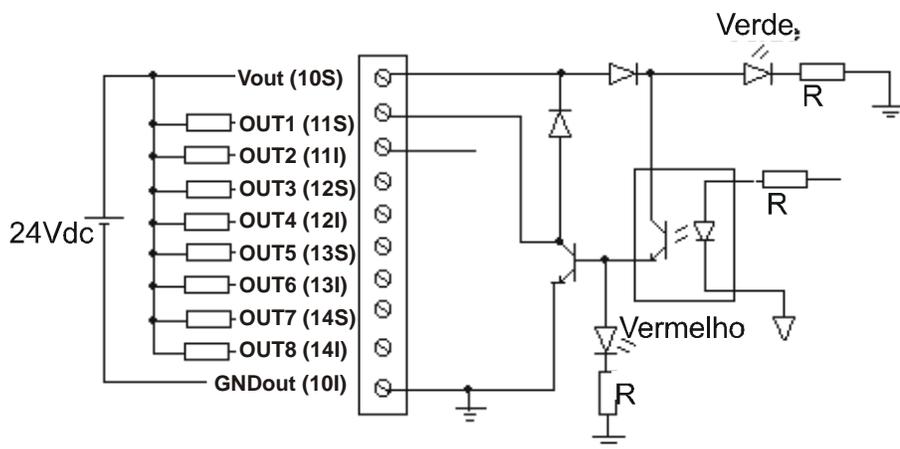



Figura 1.5 – Exemplo de conexões de saída

O **DC303** não é alimentado via barramento. O **DC303** usa a taxa de 31,25 Kbit/s em modo de tensão para a sinalização física. Muitos tipos de equipamentos Profibus-PA podem ser conectados no mesmo barramento, podendo ser alimentados por este ou não. Quando alimentados, devem usar a mesma sinalização. A quantidade de equipamentos em uma rede Profibus-PA depende da classificação de área, consumo de cada equipamento e total no barramento, distâncias envolvidas,etc.

O **DC303** não é alimentado pelo barramento.

Em áreas perigosas, o número de equipamentos deve ser limitado às restrições de segurança intrínseca.

O **DC303** é protegido contra polaridade reversa e pode suportar até ±35 Vdc sem danos.

NOTA
Por favor, refira-se ao Manual de instalação Geral do Profibus para maiores detalhes.

ATENÇÃO

ÁREAS PERIGOSAS

Em áreas perigosas que exigem segurança intrínseca ou cuidada em relação a explosões, as entidades de circuito e instalações devem ser observadas.

O acesso dos cabos de sinal aos terminais de ligação pode ser feito utilizando-se eletrodutos e conduítes.

Se outras certificações forem necessárias, refira-se ao certificado ou à norma específica para as restrições de instalação.

Topologia e Configuração da rede

A topologia em Barramento (Veja Figura 1.6 – Topologia Barramento) e topologia em Árvore (Veja Figura 1.7 – Topologia Árvore) são suportadas. Ambos os tipos possuem um barramento principal com dois terminadores. Os equipamentos são conectados ao tronco principal através das derivações (braços).

Em uma derivação podem ser conectados mais de um equipamento, dependendo do comprimento da mesma.

Acopladores ativos podem ser usados para se estender o comprimento da derivação.

O comprimento total do cabeamento, incluindo as derivações entre dois equipamentos não deve exceder a 1900 m.

ATENÇÃO

FONTES DE ALIMENTAÇÃO

Onde se requer isolação entre as fontes das entradas e saídas, recomenda-se usar no mínimo duas fontes de alimentação, sendo uma para as entradas e outra para as saídas e Vdc.

Onde a aplicação não exigir isolação entre as fontes, pode-se ter somente uma fonte para entradas, saídas e Vdc.

As saídas e entradas são isoladas opticamente.

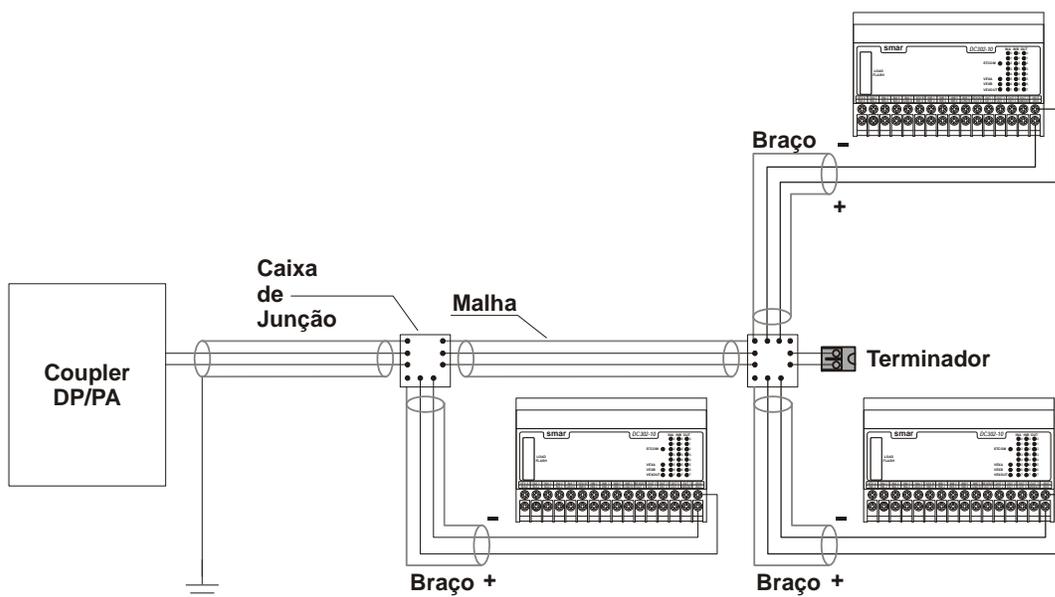



Figura 1.6- Topologia Barramento

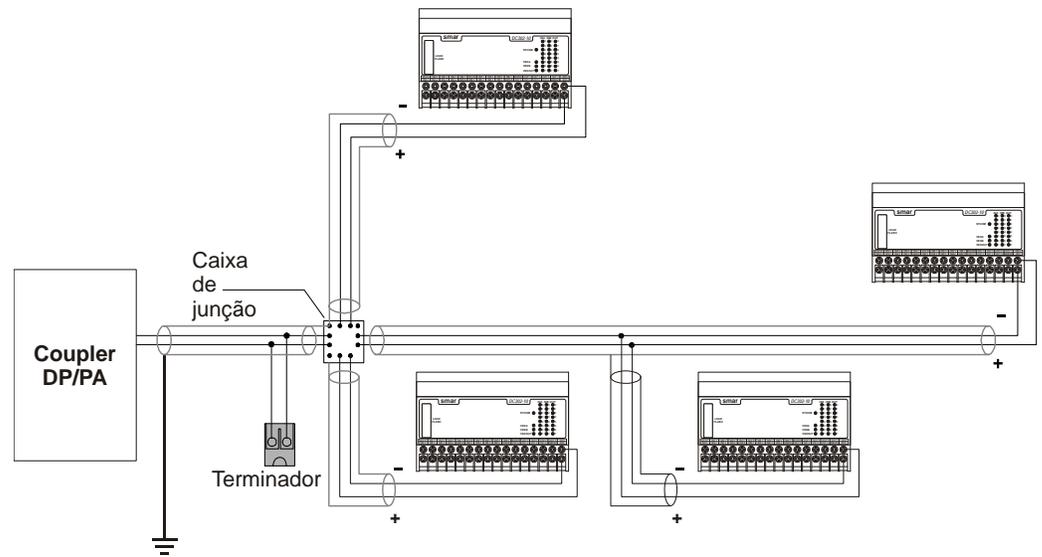



Figura 1.7 – Topologia Árvore

Sistema Geral

De acordo com a figura a seguir, veja uma topologia de rede genérica onde o **DC303** é integrado em uma rede Profibus simples.

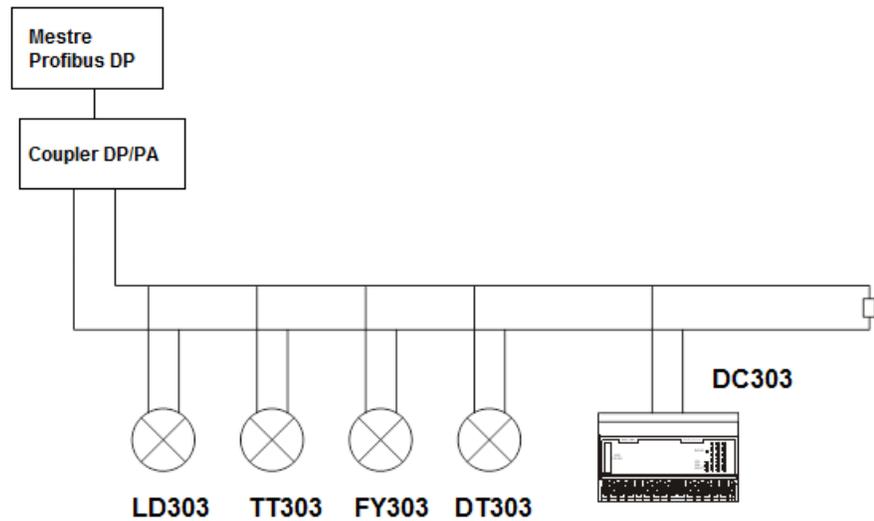



Figura 1.8 – DC303 e um Sistema Genérico Profibus

OPERAÇÃO

O **DC303** recebe até 16 entradas isoladas oticamente e pode acionar até 8 saídas em coletor aberto, de tal forma a interfacear pontos discretos ao barramento Profibus-PA. Possui ainda um bloco Flexível built-in para execução de lógicas com recursos booleanos, timers, contadores, flip-flops, etc.

Blocos funcionais fornecem grande flexibilidade em estratégias de controle.

As E/S discretas convencionais trabalham junto com os equipamentos puramente Profibus-PA integrados numa mesma rede e numa mesma malha de controle.

Blocos Funcionais de Saídas em caso de falhas incluem procedimentos padrões de mecanismo de segurança de acordo com o padrão Profibus-PA.

Entradas e saídas são isoladas umas das outras e acessadas via rede de comunicação através dos canais dos blocos funcionais. Os LEDs são utilizados para indicar o estado das entradas e saídas. O uso dos Blocos Funcionais Profibus-PA torna o sistema homogêneo de tal forma que equipamento de entradas e saídas discretas e analógicas convencionais possam estar disponíveis para facilitar a configuração de estratégias de controle, parecendo como simples equipamentos em um barramento Profibus-PA.

Descrição Funcional – Eletrônica

Veja o Diagrama de Blocos (Figura 2.1 – *Diagrama de Blocos **DC303***). A função de cada bloco é descrita a seguir.

Unidade Central de Processamento (CPU), RAM, FLASH, EEPROM

A CPU é a parte inteligente do **DC303**, sendo responsável pelo gerenciamento e operação do bloco de execução, autodiagnose e comunicação. O programa é armazenado em uma memória Flash e os dados temporários em uma memória RAM. Na falta de energia os dados armazenados na RAM são perdidos. A memória EEPROM armazena os dados não-voláteis que serão usados posteriormente. Exemplos de tais dados são: calibração, configuração e dados de identificação.

Controlador da Comunicação

É responsável pela monitoração da atividade da linha, modulação e demodulação dos sinais do barramento.

Fonte de Alimentação

Alimenta os circuitos do **DC303**.

Inicialização de Fábrica (Factory Reset)

O **DC303** possui dois contatos no lado superior esquerdo do seu invólucro (onde pode ser visto a inscrição de factory reset) que permitem a inicialização de fábrica. Para efetuar esta operação, desligue o equipamento e ligue-o fazendo um curto-circuito nestes contatos, mantendo-o até que o led de saving seja ativado.

Latches de Entrada

São latches que armazenam as condições das entradas.

Latches de saída

São latches que armazenam as condições das saídas.

Isolação Ótica

Isolação Ótica para as entradas e saídas.

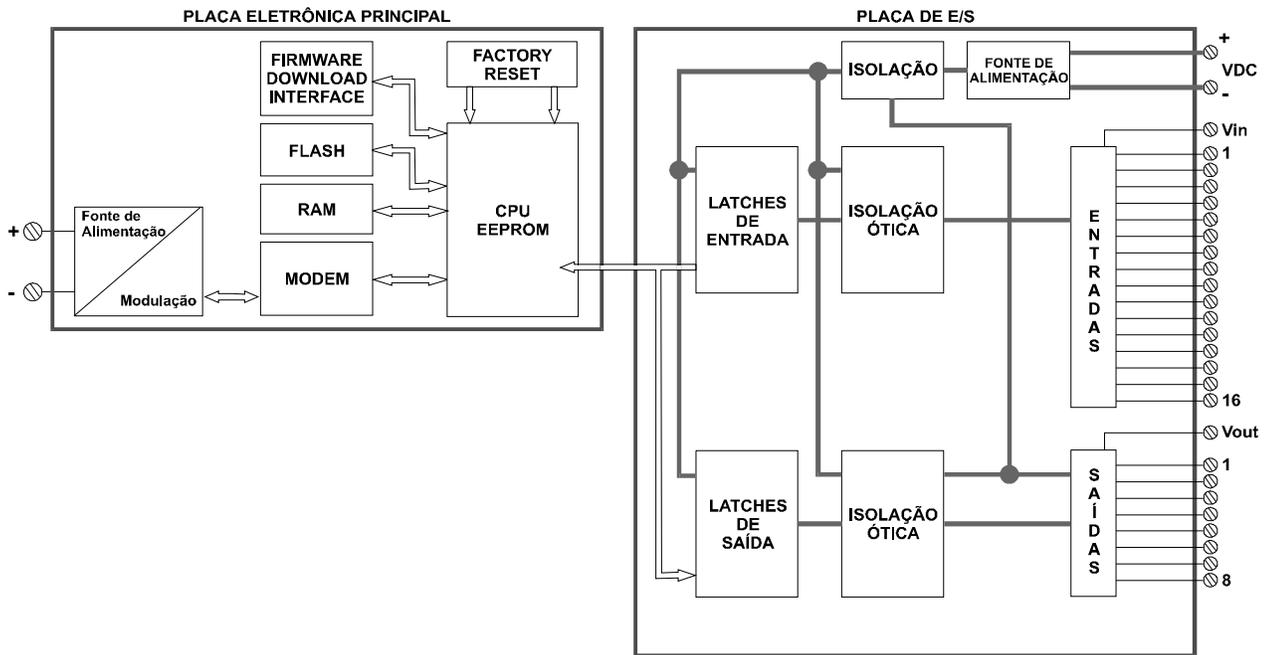



Figura 2.1 – Diagrama de Blocos DC303

CONFIGURAÇÃO

O **DC303** pode ser configurado via SYSTEM302 ou qualquer outra ferramenta segundo os padrões Profibus usando EDDL ou FDT/DTM.

O **DC303** possui os seguintes Blocos Funcionais Discrete Input e Discrete Output. Possui ainda um Bloco Flexível built-in para execução de lógicas com recursos booleanos, timers, contadores, etc.

Os Blocos Funcionais não são citados neste manual. Para explicações e detalhes, refira-se ao manual de Blocos Funcionais.

Conexão Física ao Bloco DI (Entrada Digital)

O Bloco DI utiliza um dado discreto de entrada, selecionado via canal e o deixa disponível para outro bloco funcional através de sua saída.
Para maiores informações e detalhes, refira-se ao manual dos Blocos Funcionais.

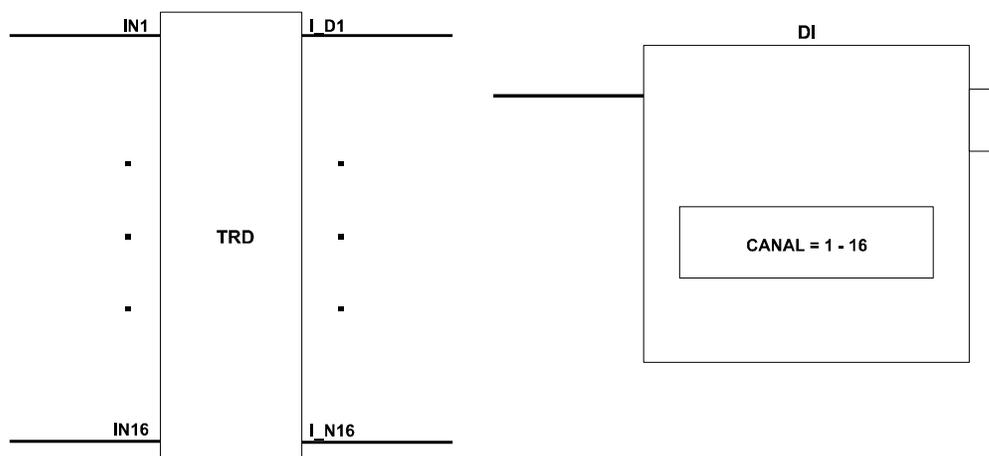



Figure 3.1 – O DC303 e as conexões com o Bloco DI.

Conexão Física ao Bloco DO (Saída Digital)

O bloco funcional DO converte o valor de SP_D para um valor útil ao hardware, através do canal selecionado.
Para maiores informações e detalhes, por favor, referencie-se ao manual de Blocos Funcionais.

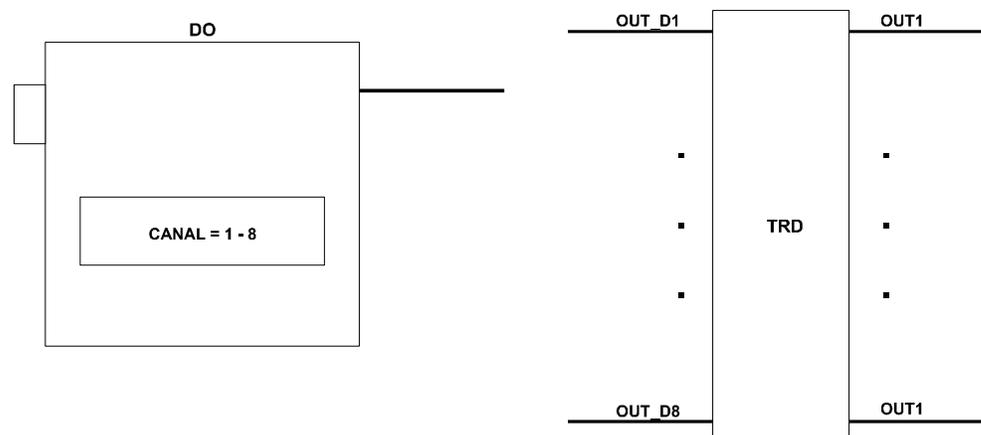



Figure 3.2 - O DC303 e as conexões com o Bloco DO.

Exemplos de Aplicações

Aplicação 1: Um computador pode manipular as entradas e saídas.

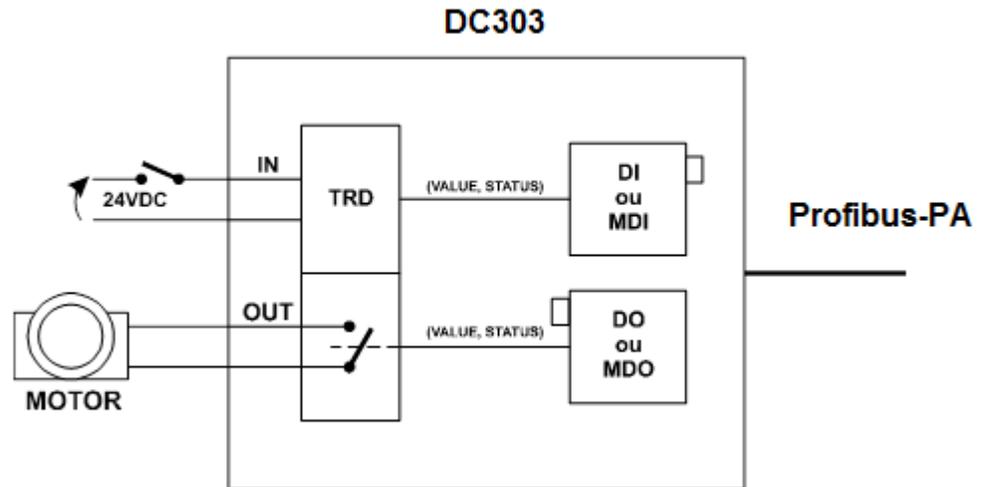



Figure 3.3 - Aplicação 1- DC303

Aplicação 2: Controle de Nível (O limite de nível baixo acionará um motor, bomba ou uma válvula on/off).

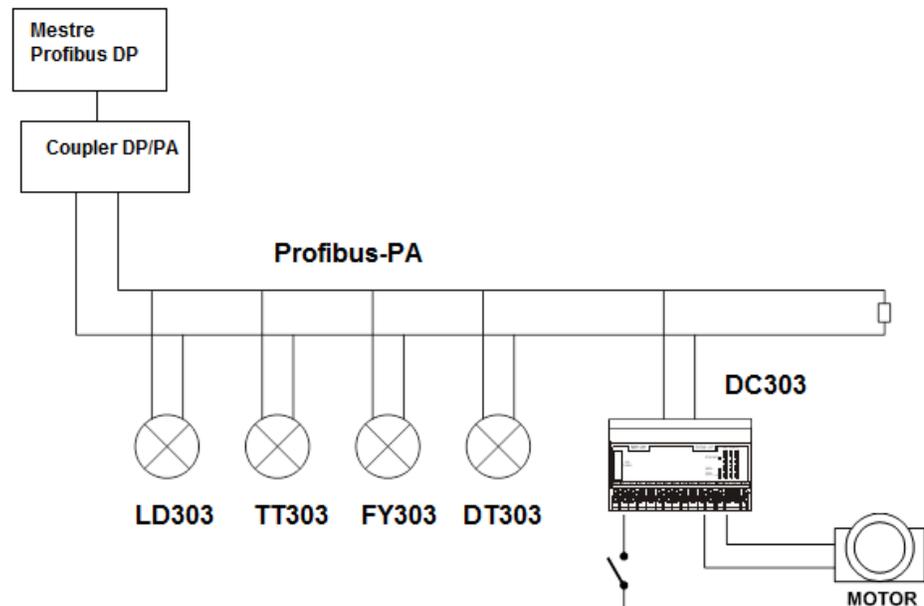



Figure 3.4 - Aplicação 2 - DC303

Aplicação 3: Aplicação genérica para o **DC303**.

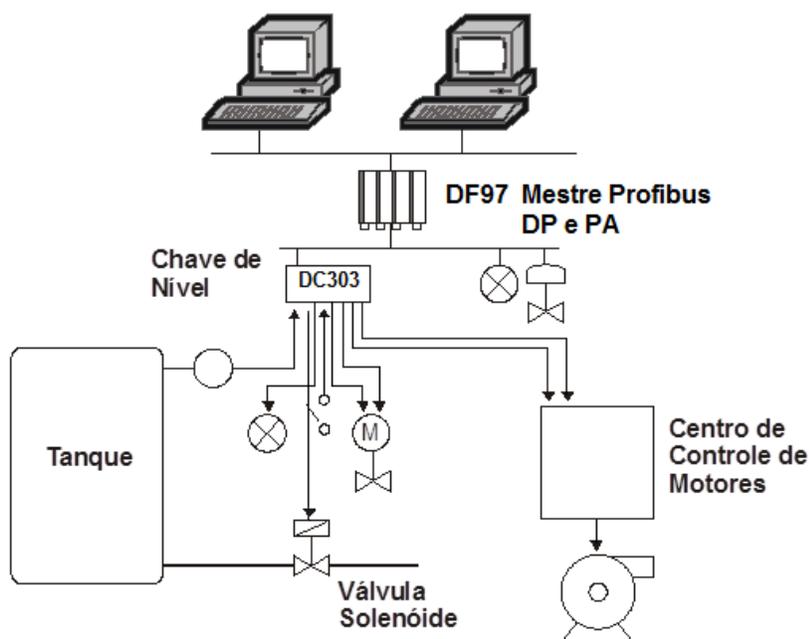



Figure 3.5 - Aplicação 5 - DC303

Execução de Lógicas no DC303

Descrição

O **DC303** foi projetado para executar lógicas. Neste caso os blocos de saídas discretas DOs não atuarão fisicamente no hardware. O bloco transdutor possui um bloco funcional flexivel built-in (FFB) que pode receber até 8 entradas discretas vindas da rede Profibus via parâmetros SP_D dos blocos DOs. No bloco transdutor estes parâmetros são denominados de IN_D1 a IN_D8. O bloco transdutor também pode disponibilizar 8 saídas discretas à rede Profibus, através dos parâmetros OUT_D1 a OUT_D8 que são disponibilizados via blocos de entradas discretas DIs (DI1 a DI8). Pode receber até 16 entradas discretas via hardware e também disponibilizar 8 saídas discretas de hardware. Nesta situação os blocos DOs e DIs devem estar em automático (mode block). E ainda, automaticamente ao se habilitar o bloco FFB no bloco transdutor (via parâmetro TRD_FFB_ENABLED), os blocos DI9 a DI16 são colocados em "Out of Service".

Quando o bloco FFB estiver desabilitado, o **DC303** funciona com 16 blocos DIs e 8 blocos DOs, lendo suas 16 entradas discretas de hardware e atuando em suas 8 saídas de hardware, respectivamente.

A indicação do estado (status) das entradas depende do sub-sistema de E/S.

O Bloco FFB provê lógicas como AND, OR, XOR e NOT e funções como: Timer On-Delay, Timer Off-Delay, Timer Pulse, Pulse Counter Down (CTD), Pulse Counter Up (CTU), Flip-Flop RS e Flip-Flop SR. As lógicas são feitas utilizando-se as entradas discretas (IN_Dx) vindas da rede Profibus via DOs (SP_D), as saídas disponíveis a rede Profibus (OUT_Dx via DIs), as entradas discretas de hardware, as saídas discretas de hardware, valores discretos de segurança em condições de falha (FSx) e variáveis discretas auxiliares (AUX's).

Status

O status das saídas OUT_Dx será de acordo com:

- Falha de entrada – ruim: Equipamento em falha (Input failure – Bad: Device Failure);
- Partida – ruim: equipamento em falha (Power up – Bad: Device Failure).

Na execução da lógica, um status maior ou igual a 0x80 é considerado "verdadeiro" (true) e menor que 0x80, é considerado "falso" (false).

Modos Suportados

Bloco Transdutor: O/S e AUTO.
As mudanças nas Linhas Lógicas e seus parâmetros de configuração dependem da seleção de CHANGE_OPTION.

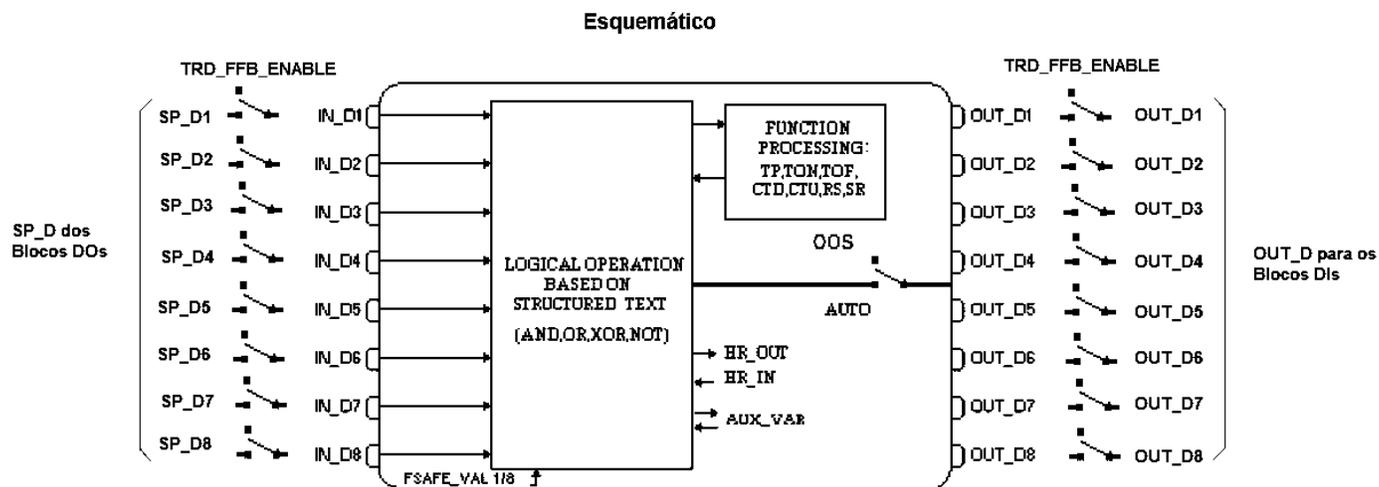



Figura 3.6 - Esquemático

Parâmetros

Idx	Parâmetro	Tipo Dado (comp)	Faixa Válida/ Opções	Valor Default	Unid.	Armaz.	Descrição
16	TRD_FFB_ENABLE	Unsigned8	0- Disabled ; 1- Enabled	0- Disabled		S	Permite a execução do FFB no TRD Block.
17	IN_D1	DS-34				D	Entrada discreta nº 1 utilizada pelo cálculo do bloco.Vem do DO_1(SP_D)
18	IN_D2	DS-34				D	Entrada discreta nº 2 utilizada pelo cálculo do bloco.Vem do DO_2 (SP_D)
19	IN_D3	DS-34				D	Entrada discreta nº 3 utilizada pelo cálculo do bloco.Vem do DO_3 (SP_D)
20	IN_D4	DS-34				D	Entrada discreta nº 4 utilizada pelo cálculo do bloco.Vem do DO_4 (SP_D)
21	IN_D5	DS-34				D	Entrada discreta nº 5 utilizada pelo cálculo do bloco.Vem do DO_5 (SP_D)
22	IN_D6	DS-34				D	Entrada discreta nº 6 utilizada pelo cálculo do bloco.Vem do DO_6 (SP_D)
23	IN_D7	DS-34				D	Entrada discreta nº 7 utilizada pelo cálculo do bloco.Vem do DO_7 (SP_D)
24	IN_D8	DS-34				D	Entrada discreta nº 8 utilizada pelo cálculo do bloco. Vem do DO_8(SP_D).
25	FSTATE_VAL_D1	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 1.
26	FSTATE_VAL_D2	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 2.
27	FSTATE_VAL_D3	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 3.
28	FSTATE_VAL_D4	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 4.
29	FSTATE_VAL_D5	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 5.
30	FSTATE_VAL_D6	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 6.
31	FSTATE_VAL_D7	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 7.
32	FSTATE_VAL_D8	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída

Idx	Parâmetro	Tipo Dado (comp)	Faixa Válida/ Opções	Valor Default	Unid.	Armaz.	Descrição
							de hardware nº 8.
33	OUT_D1	DS-34				D	Variável discreta da saída 1 calculada pelo bloco quando em modo AUTO e copiada para o DI_1.
34	OUT_D2	DS-34				D	Variável discreta da saída 2 calculada pelo bloco quando em modo AUTO e copiada para o DI_2.
35	OUT_D3	DS-34				D	Variável discreta da saída 3 calculada pelo bloco quando em modo AUTO e copiada para o DI_3.
36	OUT_D4	DS-34				D	Variável discreta da saída 4 calculada pelo bloco quando em modo AUTO e copiada para o DI_4.
37	OUT_D5	DS-34				D	Variável discreta da saída 5 calculada pelo bloco quando em modo AUTO e copiada para o DI_5.
38	OUT_D6	DS-34				D	Variável discreta da saída 6 calculada pelo bloco quando em modo AUTO e copiada para o DI_6.
39	OUT_D7	DS-34				D	Variável discreta da saída 7 calculada pelo bloco quando em modo AUTO e copiada para o DI_7.
40	OUT_D8	DS-34				D	Variável discreta da saída 8 calculada pelo bloco quando em modo AUTO e copiada para o DI_8.
41	AUX_01_16	Bitstring(2)				D/ OS	Variável Auxiliar bit enumerated 01_16.
42	AUX_17_32	Bitstring(2)				D/ OS	Variável Auxiliar bit enumerated 17_32.
43	AUX_33_48	Bitstring(2)				D/ OS	Variável Auxiliar bit enumerated 33_48.
44	AUX_49_64	Bitstring(2)				D/ OS	Variável Auxiliar bit enumerated 49_64.
45	AUX_65_80	Bitstring(2)				D/ OS	Variável Auxiliar bit enumerated 65_80.
46	AUX_81_96	Bitstring(2)				D/ OS	Variável Auxiliar bit enumerated 81_96.
47	TON_PST	16 Floats	Positive	0	Seg	S/ OS	Vetor de 16 elementos em ponto flutuante onde o usuário pode setar em segundos o valor PST para cada Timer ON Delay.
48	TON_CTA	16 Floats		0	seg	D	Vetor de 16 elementos em ponto flutuante onde o usuário pode ver em segundos o valor decorrido para cada Timer ON Delay.
49	TON_OUT	Bitstring(2)				D	Variável que indica os estados dos timers ativos.
50	TOFF_PST	16 Floats	Positive	0	Seg	S/ OS	Vetor de 16 elementos em ponto flutuante onde o usuário pode setar em segundos o valor PST para cada Timer OFF Delay.
51	TOFF_CTA	16 Floats		0	seg	D	Vetor de 16 elementos em ponto flutuante onde o usuário pode ver em segundos o valor decorrido para cada Timer OFF Delay.
52	TOFF_OUT	Bitstring(2)				D	Variável que indica os estados dos timers ativos.
53	TP_PST	16 Floats	Positive	0	Seg	S/ OS	Vetor de 16 elementos em ponto flutuante onde o usuário pode setar em segundos o valor PST para cada Timer PULSE.
54	TP_CTA	16 Floats		0	Seg	D	Vetor de 16 elementos em ponto flutuante onde o usuário pode ver em segundos o valor decorrido para cada Timer PULSE.
55	TP_OUT	Bitstring(2)				D	Variável que indica os estados dos timers ativos.
56	CTU_PST	16 Unsigned32	Positive	0	Nenhuma	S/ OS	Vetor de 16 elementos unsigned integer32 onde pode-se configurar o valor a ser contado para cada contador. O contador irá incrementar de zero ao valor PST.
57	CTU_CTA	16 Unsigned32		0	Nenhuma	D	Vetor de 16 elementos unsigned integer32 onde pode-se acompanhar a contagem parcial para cada contador.
58	CTU_OUT	Bitstring(2)				D	Variável que indica os estados dos contadores ativos.
59	CTD_PST	16 Unsigned32	Positive	0	Nenhuma	S/ OS	Vetor de 16 elementos unsigned integer32 onde pode-se configurar o valor a ser contado para cada contador. O contador irá decrementar do valor PST até zero.
60	CTD_CTA	16 Unsigned32		0	Nenhuma	D	Vetor de 16 elementos unsigned integer32 onde pode-se acompanhar a contagem parcial para cada contador.
61	CTD_OUT	Bitstring(2)				D	Variável que indica os estados dos timers ativos.
62	RS_OUT	Bitstring(2)				D	Variável que indica os estados dos Flip-Flops RS ativos.
63	SR_OUT	Bitstring(2)				D	Variável que indica os estados dos Flip-Flops SR ativos.
64	LOGIC_01	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 1.
65	LOGIC_02	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 2.

Idx	Parâmetro	Tipo Dado (comp)	Faixa Válida/ Opções	Valor Default	Unid.	Armaz.	Descrição
66	LOGIC_03	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 3.
67	LOGIC_04	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 4.
68	LOGIC_05	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 5.
69	LOGIC_06	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 6.
70	LOGIC_07	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 7.
71	LOGIC_08	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 8.
72	LOGIC_09	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 9.
73	LOGIC_10	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 10.
74	LOGIC_11	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 11.
75	LOGIC_12	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 12.
76	LOGIC_13	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 13.
77	LOGIC_14	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 14.
78	LOGIC_15	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 15.
79	LOGIC_16	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 16.
80	LOGIC_17	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 17.
81	LOGIC_18	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 18.
82	LOGIC_19	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 19.
83	LOGIC_20	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 20.
84	LOGIC_21	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 21.
85	LOGIC_22	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 22.
86	LOGIC_23	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 23.
87	LOGIC_24	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 24.
88	LOGIC_25	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 25.
89	LOGIC_26	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 26.
90	LOGIC_27	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 27.
91	LOGIC_28	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 28.
92	LOGIC_29	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 29.
93	LOGIC_30	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 30.
94	LOGIC_31	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 31.
95	LOGIC_32	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 32.
96	LOGIC_33	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 33.
97	LOGIC_34	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 34.
98	LOGIC_35	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 35.
99	LOGIC_36	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 36.
100	LOGIC_37	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 37.
101	LOGIC_38	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 38.
102	LOGIC_39	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 39.
103	LOGIC_40	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 40.
104	LOGIC_41	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 41.
105	LOGIC_42	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 42.
106	LOGIC_43	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 43.
107	LOGIC_44	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 44.
108	LOGIC_45	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 45.
109	LOGIC_46	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 46.
110	LOGIC_47	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 47.
111	LOGIC_48	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 48.
112	LOGIC_49	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 49.
113	LOGIC_50	VisibleString(24)		Espaços	Na	S/ OS	Linha Lógica de comando 50.
114	LOGIC_CHECK	Unsigned8	0 - Enable., 1 - Checked. 2- Changed but not checked yet.	1 - Checked.	Na	D/OS	Permite a verificação das linhas lógicas.
115	ERROR_LINE	Unsigned8	1-50	1	Na	S	Indica a linha onde se tem erro. Valor igual a zero indica que não existe linha com erro.
116	ERROR_CODE	Unsigned8	0 - Logic Ok. 1 - Exceed String Length or string not valid. 2 - Non valid operand. 3 - No implemented logic or missing ';' 4 - Missing parentheses or argument not valid. 5 - Non valid resource. 6 - Argument not valid. 7 - Function not valid 8 - Non available resource. 9 - Non valid attribution. 10 - First Argument not valid. 11- Second Argument not valid.	3 - No implemented logic or missing ';'	Na	S	Indica o código de erro.
117	CHANGE_OPTION	Unsigned8	0 - Logic parameter changes are only allowed in Out of Service. 1 - Always accept Logic parameter changes.	0 - Logic parameter changes are only allowed in Out of Service.	Na	S	Habilita mudanças nas linhas de lógicas independentes do modo de operação do bloco.

A tabela, a seguir, descreve as Operações Lógicas e os Comandos de Linha e seus Símbolos Correspondentes usados nas linhas lógicas:

Operação Lógica e Comando de Linha	Símbolo - descrição
E	&
OU	|
OU Exclusivo	^
Inversor	!
Igual	=
(arg1,arg2)	Argumentos de função
;	Finalizador de linha

A lógica Inversora (!) trabalha somente com variáveis simples. Exemplo: OUT1=!IN1;

Note que não é permitido ter, por exemplo, OUT1=!TP01(IN1);. Para se trabalhar desta forma, deve-se fazer: A01= TP01(IN1);. -> OUT1=!A01;

A execução da linha lógica é sempre executada linha por linha e da esquerda para a direita. Não são permitidos espaços entre caracteres. **Não são permitidas linhas vazias entre linhas com lógicas implementadas, isto é, a implementação deve ser em sequência. Nas linhas não usadas deve-se digitar ; (ponto e vírgula).**

Depois de escrever a lógica dentro dos parâmetros LOGIC_XX (XX:01 -> XX:50), o usuário precisa selecionar a opção “Enable” no parâmetro LOGIC_CHECK para verificar se há erros de sintaxe. **Quando se utiliza o processo de download, é primordial configurar, primeiramente, os parâmetros LOGIC_XX (XX:01 -> XX:50) e depois o parâmetro LOGIC_CKECK. Esta sequência é fundamental para executar a verificação.**
A tabela abaixo mostra o mnemônico para cada parâmetro de bloco usado nas linhas lógicas. O mnemônico deve estar em letras maiúsculas:

Parâmetro	Mnemônico
HW_IN.Value1	I01
HW_IN.Value2	I02
HW_IN.Value3	I03
HW_IN.Value4	I04
HW_IN.Value5	I05
HW_IN.Value6	I06
HW_IN.Value7	I07
HW_IN.Value8	I08
HW_IN.Value9	I09
HW_IN.Value10	I10
HW_IN.Value11	I11
HW_IN.Value12	I12
HW_IN.Value13	I13
HW_IN.Value14	I14
HW_IN.Value15	I15
HW_IN.Value16	I16
HW_IN.Status	SI
HW_OUT.Status	SO
HW_OUT.Value1	O1
HW_OUT.Value2	O2
HW_OUT.Value3	O3
HW_OUT.Value4	O4
HW_OUT.Value5	O5
HW_OUT.Value6	O6
HW_OUT.Value7	O7

Parâmetro	Mnemônico
HW_OUT.Value8	O8
IN_D1.Status	IN1S
IN_D2.Status	IN2S
IN_D3.Status	IN3S
IN_D4.Status	IN4S
IN_D5.Status	IN5S
IN_D6.Status	IN6S
IN_D7.Status	IN7S
IN_D8.Status	IN8S
IN_D1.Value	IN1
IN_D2.Value	IN2
IN_D3.Value	IN3
IN_D4.Value	IN4
IN_D5.Value	IN5
IN_D6.Value	IN6
IN_D7.Value	IN7
IN_D8.Value	IN8
OUT_D1.Status	SOUT1
OUT_D2.Status	SOUT2
OUT_D3.Status	SOUT3
OUT_D4.Status	SOUT4
OUT_D5.Status	SOUT5
OUT_D6.Status	SOUT6
OUT_D7.Status	SOUT7
OUT_D8.Status	SOUT8
OUT_D1.Value	OUT1
OUT_D2.Value	OUT2
OUT_D3.Value	OUT3
OUT_D4.Value	OUT4
OUT_D5.Value	OUT5
OUT_D6.Value	OUT6
OUT_D7.Value	OUT7
OUT_D8.Value	OUT8
FSTATE_VAL_D1	FS1
FSTATE_VAL_D2	FS2
FSTATE_VAL_D3	FS3
FSTATE_VAL_D4	FS4
FSTATE_VAL_D5	FS5
FSTATE_VAL_D6	FS6
FSTATE_VAL_D7	FS7
FSTATE_VAL_D8	FS8
AUX_01_16	A01-A16
AUX_17_32	A17-A32
AUX_33_48	A33-A48
AUX_49_64	A49-A64
AUX_65_80	A65-A80
AUX_81_96	A81-A96
TON	TON01-TON16
TOFF	TOF01-TOF16

Parâmetro	Mnemônico
TP	TP01-TP16
CTU	CTU01-CTU16
CTD	CTD01-CTD16
RS	RS01-RS16
SR	SR01-SR16

Funções

Para cada tipo de função há 16 recursos disponíveis e o usuário pode usar somente um recurso de cada vez. O usuário poderá atribuir o resultado de uma função aos bits discretos auxiliares.

TP TIMER PULSE

Esta função gera em sua saída um pulso de largura de tempo fixa a cada transição de falso para verdadeiro (borda de subida) na entrada. A largura de pulso pode ser programada no parâmetro TP_PST, em segundos. Transições na entrada serão ignoradas, enquanto a saída estiver ativa. O timer corrente pode ser monitorado no parâmetro TP_CTA.

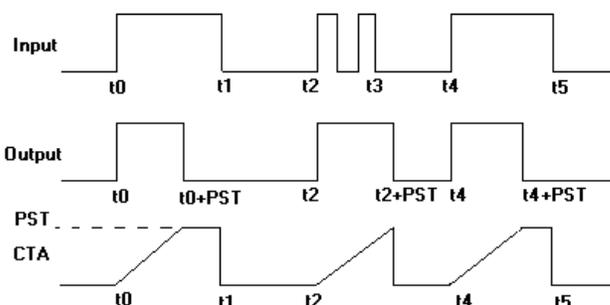



Figura 3.7 - Função Timer Pulse – diagramas de tempo

A sintaxe para a função Timer Pulse é: TPxx(arg)
Onde, xx é o recurso de 01 a 16 e arg é o argumento da função e deve ser uma variável simples. Exemplos:

O1=TP01(IN1);
OUT1= TP01(A05);
OUT3=TP08(FS1);

Por exemplo, as lógicas abaixo não são permitidas, pois os argumentos são resultados de uma função:

O1=TP01(IN1&IN2);: note que o argumento é o resultado de uma operação e isto não é permitido.

O1=TP10(!IN1);: note que o argumento é o resultado de função NOT e isto não é permitido.

O1=TP10(CTD01(IN1,IN2));: note que o argumento é o resultado de uma função e isto não é permitido.

TON TIMER ON-DELAY

Nesta função a saída vai para verdadeiro após um período de tempo em segundos configurável no parâmetro TP_PST, após a entrada ir para verdadeiro. Se a entrada for para falso antes do tempo PST, a saída permanecerá em falso. O parâmetro CTA mostra o tempo decorrente até o valor PST.

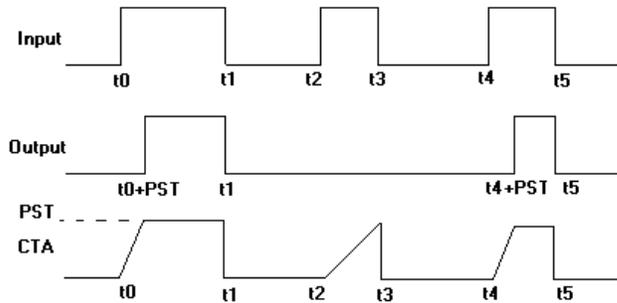



Figura 3.8 - Função Timer On-Delay– diagramas de tempo

A sintaxe para a função Timer On-Delay é: TONxx(arg)
Onde, xx é o recurso de 01 a 16 e arg é o argumento da função e deve ser uma variável simples. Exemplos:

O1=TON01(IN1)&SI;
OUT1= TON01(A05);
OUT3=TON08(FS1);

Por exemplo, as lógicas abaixo não são permitidas, pois os argumentos são resultados de uma função:

O1=TON01(IN1&IN2);: note que o argumento é o resultado de uma operação e isto não é permitido.

O1=TON10(!IN1);: note que o argumento é o resultado de uma função NOT e isto não é permitido.

O1=TON10(CTD01(IN1,IN2));: note que o argumento é o resultado de uma função e isto não é permitido.

TOF TIMER OFF-DELAY

Esta função estende o estado verdadeiro da entrada por um período de tempo em segundos configurado via parâmetro TOF_PST. Se a entrada vai para verdadeiro antes da saída ir para falso, a saída ficará em verdadeiro até que a saída vá para falso e decorra o tempo PST. O parâmetro CTA mostra o tempo corrente.

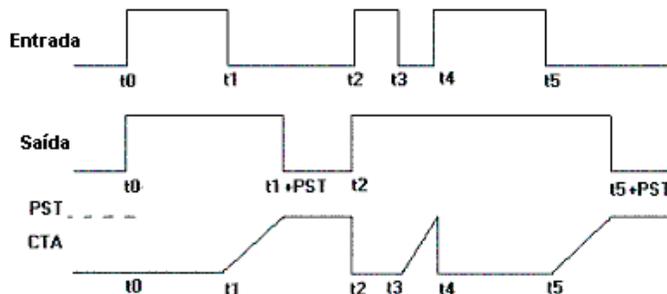



Figura 3.9 - Função Timer OFF-Delay – Diagramas de Tempo

A sintaxe para Timer Off-Delay é: TOFxx(arg)
Onde, xx é o recurso usado de 01 a 16 e arg é o argumento da função e deve ser uma variável simples. Exemplos:

O1=TOF01(IN1)&SI;
OUT1= TOF01(A05);
OUT3=TOF08(FS1);

Por exemplo, as lógicas abaixo não são permitidas, pois os argumentos são resultados de uma função:

O1=TOF01(IN1&IN2);: note que o argumento é o resultado de uma operação e isto não é permitido.

O1=TOF10(!IN1);: note que o argumento é o resultado de uma função e isto não é permitido.

O1=TOF10(CTD01(IN1,IN2));: note que o argumento é o resultado de uma função e isto não é permitido.

CTD PULSE COUNTER DOWN

Esta função é usada para contar transições de subida (de falso para verdadeiro) da entrada (arg1). A cada transição de subida, o parâmetro CTA decrementa de uma unidade. Quando o CTA atinge o valor zero, a saída do contador irá para verdadeiro. O valor do contador CTA será carregado com o valor de PST. Uma transição de falso para verdadeiro no argumento 2 (arg2) coloca a saída em falso e o parâmetro CTA receberá o valor PST.

A sintaxe para CTD é: CTDxx(arg1,arg2)

Onde, xx é o recurso de 01 a 16 e arg1 e arg2 são argumentos das funções e devem ser variáveis simples. Exemplos:

O3=CTD10(IN1,IN2);
OUT1=CTD03(A11,A14)&SI;

Por exemplo, as lógicas abaixo não são permitidas, pois os argumentos são resultados de uma função:

O1=CTD01(IN1&IN2,IN3);: note que o argumento é o resultado de uma operação e isto não é permitido.

O1=CTD10(!IN1,IN3);: note que o argumento é o resultado de uma função e isto não é permitido.

O1=CTD10(TP01(IN1),IN2);: note que o argumento é o resultado de uma função e isto não é permitido.

CTU PULSE COUNTER UP

Esta função é usada para contar transições de subida (de falso para verdadeiro) na entrada (arg1). A cada transição o parâmetro CTA incrementará de uma unidade. Ao atingir o valor PST, a saída desta função irá para o estado lógico verdadeiro e o valor CTA será carregado com valor zero. Uma transição de falso para verdadeiro no argumento 2 (arg2) coloca a saída em falso e o parâmetro CTA receberá o valor zero.

A sintaxe para CTU é: CTUxx(arg1,arg2)

Onde, xx é o recurso de 01 a 16 e arg1 e arg2 são os argumentos da função e devem ser variáveis simples. Exemplos:

O3=CTU10(IN1,IN2);
OUT1=CTU03(A11,A14)&SI;

Por exemplo, as lógicas abaixo não são permitidas, pois os argumentos são resultados de uma função:

O1=CTU01(IN1&IN2,IN3);: note que o argumento é o resultado de uma operação e isto não é permitido.

O1=CTU10(!IN1,IN3);: note que o argumento é o resultado de uma função e isto não é permitido.

O1=CTU10(TP01(IN1),IN2);: note que o argumento é o resultado de uma função e isto não é permitido.

FLIP-FLOP RS

Esta função tem a seguinte tabela de operação:

R(arg1)	S(arg2)	OUT
0	0	Último estado
0	1	1
1	0	0
1	1	0

A sintaxe para RS Flip-Flop é: RSxx(arg1,arg2)
Onde, xx é o recurso de 01 a 16 e arg1 e arg2 são os argumentos da função e devem ser variáveis simples. Exemplos:

O3=RS10(IN1,IN2);
OUT1=RS03(A11,A14)&SI;

Por exemplo, as lógicas abaixo não são permitidas, pois os argumentos são resultados de uma função:

O1=RS01(IN1&IN2,IN3);: note que o argumento é o resultado de uma operação e isto não é permitido.

O1=RS10(!IN1,IN3);: note que o argumento é o resultado de uma função e isto não é permitido.

O1=RS10(TP01(IN1),IN2);: observe que o argumento é o resultado de uma função e isto não é permitido.

FLIP-FLOP SR

Esta função tem a seguinte tabela de operação:

S (arg1)	R (arg2)	OUT
0	0	Último Estado
0	1	0
1	0	1
1	1	1

A sintaxe para SR Flip-Flop é: SRxx(arg1,arg2)
Onde, xx é o recurso de 01 a 16 e arg1 e arg2 são os argumentos da função e devem ser variáveis simples. Exemplos:

O3=SR10(IN1,IN2);
OUT1=SR03(A11,A14)&SI;

Por exemplo, as lógicas abaixo não são permitidas, pois os argumentos são resultados de uma função:

O1=SR01(IN1&IN2,IN3);: o argumento é o resultado de uma operação e isto não é permitido.

O1=SR10(!IN1,IN3);: o argumento é o resultado de uma função NOT e isto não é permitido.

O1=SR10(TP01(IN1),IN2);: o argumento é o resultado de uma função e isto não é permitido.

Códigos de Erros

Alguns exemplos de condições de erro:

Error Code: "Exceed String Length or string not valid."
a) OUT1=IN1&IN2&IN2|IN4^IN5|IN6;
Note que existem 29 caracteres no texto e o máximo permitido é 24.

b) OUT1=IN1&in2;
Note que todos os mnemônicos devem estar em Letras Maiúsculas.

Error Code: "Non valid operand."
OUT1=IN1%IN2;
Note que o símbolo % não é permitido. Veja também a tabela que descreve a os Operadores Lógicos e as Linhas de Comando.

Error Code: "No implemented logic or missing ';'."
OUT1=IN1
Note que falta o terminador ";" no final da linha de lógica.

Error Code: "Missing parentheses or argument not valid."
OUT1=TP10(IN1;
Observe que falta um parênteses na função timer pulse.

Error Code: "Non valid resource."
OUT1=TP18(IN1);
Observe que há até 16 recursos para cada função.

Error Code: "Argument not valid."
OUT1=TP10(IN10);
Observe que há somente 8 entradas discretas vindas da rede PROFIBUS. IN10 não é um argumento válido.

Error Code: "Function not valid."
OUT1=TR10(IN1);
Observe que TR não é uma função válida.

Error Code: "Non available resource."
OUT1=TP10(IN1);
A03=TP10(IN7);

Observe que há 16 recursos para cada função e o recurso 10 já foi utilizado e não pode ser usado novamente em uma lógica posterior. O que pode ser feito é atribuir o resultado da função a uma variável auxiliar e esta ser usada várias vezes.
A03=TP10(IN7);

Error Code: "Non valid attribution."
IN1=IN2^TP03(IN4);
Observe que não é permitida atribuição para entradas.

Error Code: "First Argument not valid."
OUT1=CTD01(!IN1,IN2);
Observe que os argumentos devem ser necessariamente variáveis simples e não resultados de lógica ou funções.

OUT1=RS11(IN15,IN2);
Observe que o primeiro argumento da função não é um recurso válido.

Error Code: "Second Argument not valid."
a) OUT1=CTD01(IN1,!IN2);
Note que necessariamente os argumentos de função devem ser variáveis simples e não resultados de lógicas ou funções.

OUT1=RS11(IN1,IN20);
Observe que o segundo argumento da função não é um recurso válido.

Exemplos de aplicações

1) De acordo com a próxima figura, temos uma aplicação industrial onde se tem o enchimento de garrafas com fluido químico. A esteira é movimentada até que um sensor detecte sua presença. A esteira deve parar e abrir a válvula de enchimento até que seja detectado o nível pelo sensor de nível. Depois de detectar o nível, o sistema deve esperar durante 10 segundos e mover a esteira novamente até a próxima garrafa.

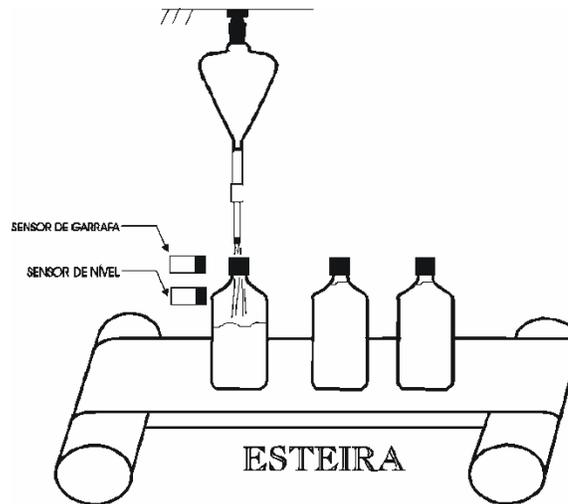



Figura 3.10 - Enchimento de garrafas com fluido químico

Usando o bloco funcional Flexível, temos as seguintes definições:

- ☑ A esteira será ligada utilizando a saída 01 de hardware (O1);
- ☑ A válvula de enchimento será ligada utilizando a saída 02 de hardware (O2);
- ☑ sensor de garrafa será conectado a entrada de 01 de hardware (I01);
- ☑ sensor de nível será conectado a entrada de 02 de hardware (I02);
- ☑ A alimentação do sistema será a entrada de 03 de hardware (I03);

Tem-se, então, a seguinte configuração:

TON_PST resource [01] = 10.0s.
LOGIC_01 A01=TON01(I02);
LOGIC_02 O1=I03&!I01|A01;
LOGIC_03 O2=I01&!I02;

Fazendo analogia com a programação ladder, tem-se:

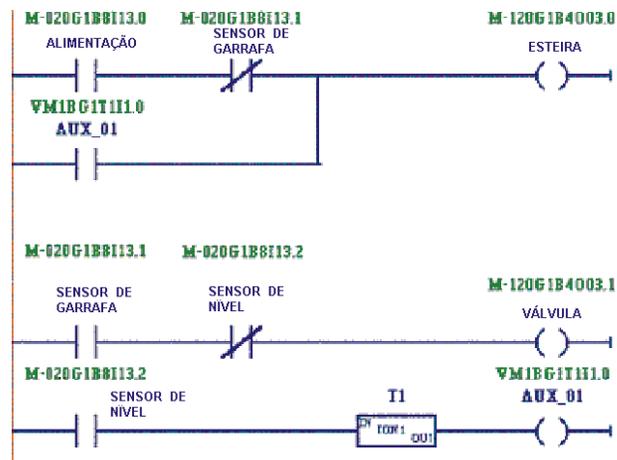



Figura 3.11 - Analogia com a programação ladder

2) Na seguinte aplicação mecânica, tem-se o controle de passos para operar uma balança eletromecânica, que efetua a pesagem de rocha fosfatada.

O peso do processo é feito por batelada, o sistema executa um ciclo completo de pesagem a cada intervalo de 20 segundos. Veja a figura a seguir:

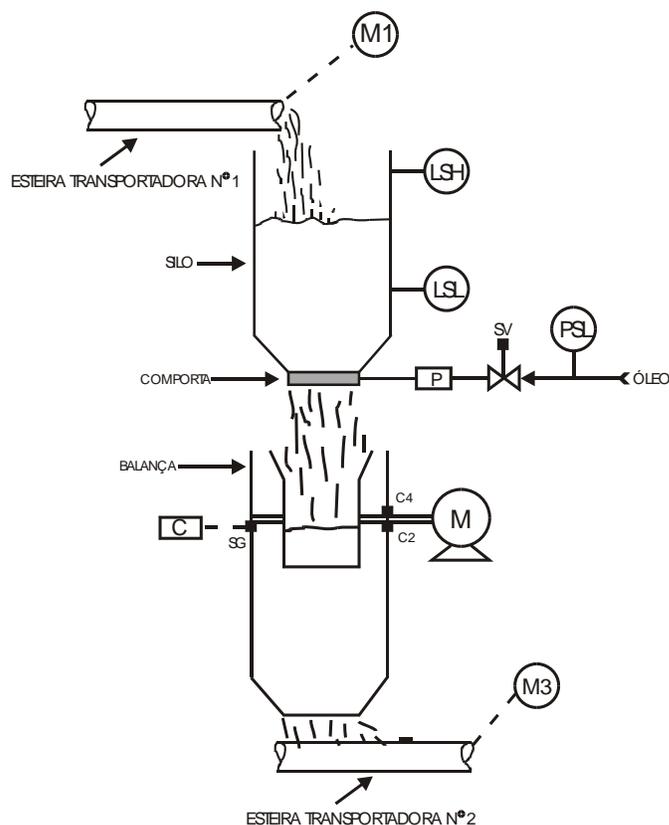



Figura 3.12 – Aplicação com balança eletromecânica

M1 e M3 - Motores acionadores das esteiras
C2 e C4 – Chaves fim de curso
LSH - Sensor de Nível Alto
LSL - Sensor de Nível Baixo
SG - Célula de Carga
SV - Válvula Solenoide
M - Motor acionador de Caçamba
P - Pistão acionador de Comporta
C - Circuito de Pesagem

Processo:

O sistema necessita das seguintes condições para fazer o startup:

- ☑ Nível da pedra fosfática (LSL desativado);
- ☑ Pressão do Óleo (PSL ligado);
- ☑ Esteira transportadora nº 2 em movimento (M3 ligado);
- ☑ Caçamba na posição inicial (C4 ligado);

Satisfeitas as condições iniciais, observa-se que:

- ☑ Acionando-se o botão de partida, a comporta abre-se, dando início ao carregamento da caçamba.
- ☑ Uma vez atingido o peso desejado, a comporta fecha. Decorridos 5 segundos a caçamba efetua um giro de 180º , descarregando o produto na esteira transportadora número 2.

Observação:

- Esta nova posição será detectada por C2 e após 5 segundos, a caçamba retornará à posição inicial e isto será detectado pelo C4.
- Depois da caçamba retornar para a posição inicial, ter-se-á um novo ciclo de pesagem.

Comentário:

- A sequência de operação deve ser interrompida se qualquer um dos pré-requisitos não for

satisfeito.
A comporta silo é acionada por um pistão hidráulico.

Usando o Bloco Funcional Flexível, tem-se as seguintes definições:

- ☑ LSL será conectado à entrada do hardware 01 (I01);
- ☑ LSH será conectado à entrada do hardware 02 (I02);
- ☑ PSL será conectado à entrada do hardware 03 (I03);
- ☑ C2 será conectado à entrada do hardware 04 (I04);
- ☑ C4 será conectado à entrada do hardware 05 (I05);
- ☑ Alimentação será conectada à entrada do hardware 06 (I06);
- ☑ M3 será conectado à entrada do hardware 07 (I07);
- ☑ M será conectado à saída do hardware 01 (O1);
- ☑ A Comport será ativada pela saída do hardware 02 (O2);
- ☑ M1 será ativado pela saída do hardware 03 (O3);

Tem-se a seguinte configuração:

TON_PST recurso [01] = 5.0s.
LOGIC_01 A01=!I01&I03&I07&I05;
LOGIC_02 A02=I06&RS01(I02,I01);
LOGIC_03 O3=A02&I03;
LOGIC_04 A03=I03&I07;
LOGIC_05 O2=I06&A03&!I04;
LOGIC_06 O1=TON01(I04)&!I05&A03;

3) Usando Valores de Estado de Falha:

Supondo que exista a seguinte condição:

- ☑ A01: recebe a lógica entre os status das entradas discretas 1 e 2, como esta: A01=IN1S&IN2S; quando o status é ruim, uma destas entradas, então A01=false(0), de outra forma, A01=true (1);
- ☑ FS1: é o valor de segurança de falha para O1;
- ☑ A02: é o bit que contém a lógica para O1;

Há a seguinte tabela entre FS1, A01 e A02:

FS1	A01	A02	O1
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

Então,

A03=!FS1&A01&A02;
A04=FS1&!A01&!A02;
A05=FS1&!A01&A02;
A06=FS1&A01&A02;
O1=A03|A04|A05|A06;

Profibus-PA: Configurando ciclicamente o DC303 – Remote I/O Profibus-PA

Configuração Cíclica do DC303

Os protocolos PROFIBUS-DP e PROFIBUS-PA possuem mecanismos contra falhas e erros de comunicação entre o equipamento da rede e o mestre. Por exemplo, durante a inicialização do equipamento esses mecanismos são utilizados para verificar esses possíveis erros. Após a energização (power up) do equipamento de campo (escravo) pode-se trocar dados ciclicamente com o mestre classe 1, se a parametrização para o escravo estiver correta. Estas informações são obtidas através dos arquivos GSDs (arquivos fornecidos pelos fabricantes dos equipamentos que contém suas descrições). Através dos comandos abaixo, o mestre executa todo o processo de inicialização com os equipamentos PROFIBUS-PA:

- **Get_Cfg:** carrega a configuração dos escravos no mestre e verifica a configuração da rede;
- **Set_Prm:** escreve nos parâmetros dos escravos e executa os serviços de parametrização da rede;
- **Set_Cfg:** configura os escravos de acordo com as entradas e saídas;
- **Get_Cfg:** um outro comando, onde o mestre verifica a configuração dos escravos.

Todos estes serviços são baseados nas informações obtidas dos arquivos GSDs dos escravos. O arquivo GSD do **DC303** mostra os detalhes de revisão do hardware e do software, bus timing do equipamento e informações sobre a troca de dados cíclicos. Para ter acesso a biblioteca completa de GSDs Smar, consulte: http://www.smar.com.br

O **DC303** possui 24 blocos funcionais: 16 Entradas Discretas (DI) e 8 Saídas Discretas (DO) . Possui também o módulo vazio (Empty module) para aplicações onde se quer configurar apenas alguns blocos funcionais. Deve-se respeitar a seguinte ordem cíclica dos blocos: DO_1, DO_2,…DO_8, DI_, DI_2, …DI_16. Supondo que se queira trabalhar somente com os blocos DOs, configure-os assim: DO_1, DO_2, DO_3, DO_4, DO_5, DO6, DO_7, DO_8, EMPTY_MODULE, EMPTY_MODULE, EMPTY_MODULE, EMPTY_MODULE, EMPTY_MODULE, EMPTY_MODULE, EMPTY_MODULE, EMPTY_MODULE. No entanto, se quiser trabalhar apenas com os blocos DOs e o DI_2, faça o seguinte DO_1, DO_2, DO_3, DO_4, DO_5, DO6, DO_7, DO_8, EMPTY_MODULE, DI_2, EMPTY_MODULE, EMPTY_MODULE, EMPTY_MODULE, EMPTY_MODULE, EMPTY_MODULE, EMPTY_MODULE.

A maioria dos configuradores PROFIBUS utiliza dois diretórios onde se deve ter os arquivos GSD's e BITMAP's dos diversos fabricantes. Os GSD's e BITMAPS para os equipamentos da Smar podem ser adquiridos via internet no site (https://www.smar.com), no link download.

O exemplo a seguir mostra os passos necessários para integrar o **DC303** em um sistema PA. Estes passos são válidos para todos os equipamentos da linha 303 da Smar:

- Copie o arquivo gsd do **DC303** para o diretório de pesquisa do configurador PROFIBUS, normalmente chamado de GSD;

- Copie o arquivo bitmap do **DC303** para o diretório de pesquisa do configurador PROFIBUS, normalmente chamado de BMP;

- Após escolher o mestre, defina a taxa de comunicação. Não esqueça que os couplers podem ter as seguintes taxas de comunicação: 45.45 kbits/s (Siemens), 93.75 kbits/s (P+F) e 12 Mbits/s (P+F, SK3). O link device IM157 pode ter até 12 Mbits/s;

- Acrescente o **DC303** e especifique o seu endereço no barramento;

- Escolha a configuração cíclica via parametrização com o arquivo gsd, que depende da aplicação, conforme visto anteriormente. Para cada bloco DO e DI, o **DC303** fornece ao mestre o valor da variável discreta e mais 1 byte de status que traz a informação da qualidade deste valor discreto.

- No bloco DO e DI, pode-se escolher as seguintes opções cíclicas abaixo, sendo que no caso do Bloco DO, somente uma delas é possível por bloco.

```
;Empty module
Module = "EMPTY_MODULE"          0x00 ;
EndModule
```

```
;
;
;Modules for Discrete Output Block
Module = "SP_D"                     0xA1 ;

EndModule

Module = "SP_D+RB_D"                0xC1, 0x81, 0x81, 0x83 ;

EndModule
Module = "SP_D+CB_D"                0xC1, 0x81, 0x82, 0x92 ;

EndModule
Module = "SP_D+RB_D+CB_D"           0xC1, 0x81, 0x84, 0x93 ;

EndModule
Module = "RIN_D+ROUT_D"             0xC1, 0x81, 0x81, 0x8C ;

EndModule
Module = "RIN_D+ROUT_D+CB_D"        0xC1, 0x81, 0x84, 0x9C ;

EndModule
Module = "SP_D+RB_D+RIN_D+ROUT_D+CB_D"        0xC1, 0x83, 0x86, 0x9F ;

EndModule

;Modules for Discrete Input Block

Module = "OUT_D"                    0x91 ;

EndModule
```

- Permite ativar a condição de watchdog, que faz o equipamento ir para uma condição de falha segura ao detectar uma perda de comunicação entre o equipamento escravo e o mestre.

Se os bloco DOs estiverem em AUTO, então o equipamento receberá o valor e status do setpoint discreto do master classe 1 e ainda o usuário poderá escrever neste valor via master classe 2. Neste caso, o status do setpoint deve ser sempre igual a 0x80 (“good”) e pode-se escolher as seguintes configurações:

- SP_D
- SP_D+RB_D
- SP_D+RB_D+CB_D

Se o bloco DOs estiverem em RCAS, o equipamento receberá o valor e status do setpoint discreto somente via master classe 1, sendo o status sempre igual a 0xc4 (“IA”). Pode-se escolher as seguintes configurações:

- SP_D
- SP_D+RB_D
- SP_D+RB_D+CB_D
- RIN_D+ROUT_D
- RIN_D+ROUT_D+CB_D
- SP_D+RB_D+RIN_D+ROUT_D+CB_D

Diagnósticos Cíclicos

Pode-se verificar os diagnósticos ciclicamente através de leituras via mestre Profibus-DP classe 1, assim como, aciclicamente, via mestre classe 2. Os equipamentos Profibus-PA disponibilizam 04 bytes padrões via Physical Block (vide figura 3.13 e figura 3.14) e quando o bit mais significativo do 4º. Byte for “1”, estenderá o diagnóstico em mais 6 bytes. Estes bytes de diagnósticos também podem ser monitorados via ferramentas acíclicas.

Figura 3.13 – Diagnóstico Cíclicos

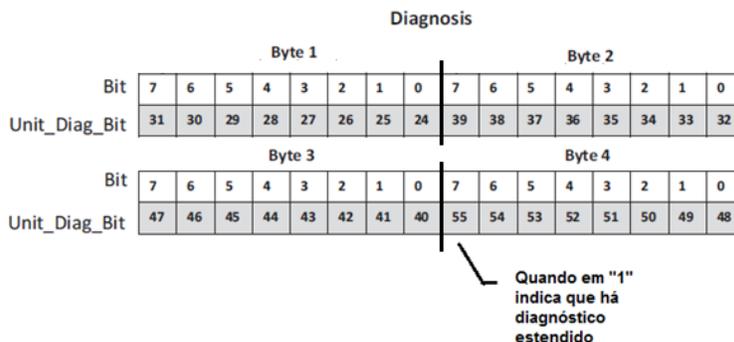



Figura 3.14 – Mapeamento dos Diagnósticos Cíclicos nos 4 bytes do Physical Block

Unit_Diag_bit está descrito no arquivo GSD do equipamento Profibus-PA.

A seguir vem parte da descrição de um arquivo GSD onde se tem os 4 bytes em detalhes:

```
;----------- Description of device related diagnosis: ---------------------
;

Unit_Diag_Bit(16)   = "Error appears"
Unit_Diag_Bit(17)   = "Error disappears"
;
;Byte 01
Unit_Diag_Bit(24)   = "Hardware failure electronics"
Unit_Diag_Bit(25)   = "Not used 25"
Unit_Diag_Bit(26)   = "Not used 26"
Unit_Diag_Bit(27)   = "Not used 27"
Unit_Diag_Bit(28)   = "Memory error"
Unit_Diag_Bit(29)   = "Not used 29"
Unit_Diag_Bit(30)   = "Device not initialized"
Unit_Diag_Bit(31)   = "Device initialization failed"

;Byte 02
Unit_Diag_Bit(32)   = "Not used 32"
Unit_Diag_Bit(33)   = "Not used 33"
Unit_Diag_Bit(34)   = "Configuration invalid"
Unit_Diag_Bit(35)   = "Restart"
Unit_Diag_Bit(36)   = "Coldstart"
Unit_Diag_Bit(37)   = "Maintenance required"
Unit_Diag_Bit(38)   = "Not used 38"
Unit_Diag_Bit(39)   = "Ident_Number violation"

;Byte 03
Unit_Diag_Bit(40)   = "Not used 40"
Unit_Diag_Bit(41)   = "Not used 41"
Unit_Diag_Bit(42)   = "Not used 42"
Unit_Diag_Bit(43)   = "Not used 43"
Unit_Diag_Bit(44)   = "Not used 44"
Unit_Diag_Bit(45)   = "Not used 45"
Unit_Diag_Bit(46)   = "Not used 46"
Unit_Diag_Bit(47)   = "Not used 47"
```

```
;byte 04
Unit_Diag_Bit(48)   = "Not used 48"
Unit_Diag_Bit(49)   = "Not used 49"
Unit_Diag_Bit(50)   = "Not used 50"
Unit_Diag_Bit(51)   = "Not used 51"
Unit_Diag_Bit(52)   = "Not used 52"
Unit_Diag_Bit(53)   = "Not used 53"
Unit_Diag_Bit(54)   = "Not used 54"
Unit_Diag_Bit(55)   = "Extension Available"

; extend diag
Unit_Diag_Bit(56)   = "Transducer Block in Out of Service"
Unit_Diag_Bit(57)   = "FFB is active in Transducer Block"
Unit_Diag_Bit(58)   = "Error Code active in FFB"
Unit_Diag_Bit(59)   = "Power Supply Failure for Inputs"
Unit_Diag_Bit(60)   = "Power Supply Failure for Outputs"
Unit_Diag_Bit(61)   = "Addr Jumper is active"
Unit_Diag_Bit(62)   = "Not used 62"
Unit_Diag_Bit(63)   = "Device is writing lock"

Unit_Diag_Bit(64)   = "Simulation Active in DI 1 and/or DI 2 Block"
Unit_Diag_Bit(65)   = "Simulation Active in DI 3 and/or DI 4 Block"
Unit_Diag_Bit(66)   = "Simulation Active in DI 5 and/or DI 6 Block"
Unit_Diag_Bit(67)   = "Simulation Active in DI 7 and/or DI 8 Block"
Unit_Diag_Bit(68)   = "Simulation Active in DI 9 and/or DI 10 Block"
Unit_Diag_Bit(69)   = "Simulation Active in DI 11 and/or DI 12 Block"
Unit_Diag_Bit(70)   = "Simulation Active in DI 13 and/or DI 14 Block"
Unit_Diag_Bit(71)   = "Simulation Active in DI 15 and/or DI 16 Block"

Unit_Diag_Bit(72)   = "Fail Safe Active in DI 1 and/or DI 2 Block"
Unit_Diag_Bit(73)   = "Fail Safe Active in DI 3 and/or DI 4 Block"
Unit_Diag_Bit(74)   = "Fail Safe Active in DI 5 and/or DI 6 Block"
Unit_Diag_Bit(75)   = "Fail Safe Active in DI 7 and/or DI 8 Block"
Unit_Diag_Bit(76)   = "Fail Safe Active in DI 9 and/or DI 10 Block"
Unit_Diag_Bit(77)   = "Fail Safe Active in DI 11 and/or DI 12 Block"
Unit_Diag_Bit(78)   = "Fail Safe Active in DI 13 and/or DI 14 Block"
Unit_Diag_Bit(79)   = "Fail Safe Active in DI 15 and/or DI 16 Block"

Unit_Diag_Bit(80)   = "Fail Safe Active in DO 1 and/or DO 2 Block"
Unit_Diag_Bit(81)   = "Fail Safe Active in DO 3 and/or DO 4 Block"
Unit_Diag_Bit(82)   = "Fail Safe Active in DO 5 and/or DO 6 Block"
Unit_Diag_Bit(83)   = "Fail Safe Active in DO 7 and/or DO 8 Block"
Unit_Diag_Bit(84)   = "Simulation Active in DO 1 and/or DO 2 Block"
Unit_Diag_Bit(85)   = "Simulation Active in DO 3 and/or DO 4 Block"
Unit_Diag_Bit(86)   = "Simulation Active in DO 5 and/or DO 6 Block"
Unit_Diag_Bit(87)   = "Simulation Active in DO 7 and/or DO 8 Block"

Unit_Diag_Bit(88)   = "DI 1 and/or DI 2 Block: Out of Service"
Unit_Diag_Bit(89)   = "DI 3 and/or DI 4 Block: Out of Service"
Unit_Diag_Bit(90)   = "DI 5 and/or DI 6 Block: Out of Service"
Unit_Diag_Bit(91)   = "DI 7 and/or DI 8 Block: Out of Service"
Unit_Diag_Bit(92)   = "DI 9 and/or DI 10 Block: Out of Service"
Unit_Diag_Bit(93)   = "DI 11 and/or DI 12 Block: Out of Service"
Unit_Diag_Bit(94)   = "DI 13 and/or DI 14 v: Out of Service"
Unit_Diag_Bit(95)   = "DI 15 and/or DI 16 Block: Out of Service"

Unit_Diag_Bit(96)   = "DO 1 Block: Out of Service"
Unit_Diag_Bit(97)   = "DO 2 Block: Out of Service"
Unit_Diag_Bit(98)   = "DO 3 Block: Out of Service"
Unit_Diag_Bit(99)   = "DO 4 Block: Out of Service"
Unit_Diag_Bit(100)  = "DO 5 Block: Out of Service"
Unit_Diag_Bit(101)  = "DO 6 Block: Out of Service"
Unit_Diag_Bit(102)  = "DO 7 Block: Out of Service"
Unit_Diag_Bit(103)  = "DO 8 Block: Out of Service"
```

Endereçamento do DC303

1. Durante o procedimento de inicialização (isto é, ao energizar o **DC303**), se o jumper Simulate estiver em “on”, o endereço é colocado nas saídas de forma que o usuário possa verificar o endereço nas saídas 1 a 7, através dos leds, sendo que o led aceso representa “1”. A Saída 1 é o bit menos significativo e a saída 7, o bit mais significativo.

2. A saída 8 é usada para sinalizar o Identifier_Number_Selector e ficará ligada(estado on) se for “ Manufacturer Specific” e apagada(estado off), se for “Profile Specific”. Quando em “Manufacturer Specific”, o Identifier Number é 0x0dca. Uma vez alterado de “Profile Specific” para “Manufacturer Specific”, deve-se esperar 5 segundos e desligar e ligar o equipamento para que o cujo identifier Number seja atualizado no nível de comunicação. Se o equipamento estiver em “Profile Specific” e com o arquivo GSD usando Identifier Number igual a 0x0dca, haverá comunicação acíclica, isto com ferramentas baseadas em EDDL, FDT/DTM, mas não haverá comunicação cíclica com o mestre Profibus-DP.

ATENÇÃO
Caso seja necessário executar o procedimento de **Reset** ou de **Desligar/Ligar**, desconecte as saídas, pois poderá haver o acionamento, uma vez que o **DC303** colocará o endereço nas saídas, se o jumper simulate estiver configurado para “on”.

Figura 3.15 - Vista superior do DC303 com detalhe do Simulation Jumper habilitado na posição ON.

3. Para se alterar o endereço pode-se utilizar qualquer ferramenta baseada em EDDL (como o Simatic PDM) ou FDT/DTM. Para isto deve-se conhecer o endereço atual do **DC303**.

Download com Simatic PDM

ATENÇÃO
No caso de se usar o Simatic PDM para configurar o **DC303**, antes de se fazer um comando de “Download to Device” deve-se fazer um comando de “**Upload to PC/PG”** e ainda, deve-se configurar o parâmetro CHANGE_OPTION com a opção " Always accept Logic parameter changes.

Seção 4

PROCEDIMENTOS DE MANUTENÇÃO

Geral

O **DC303 E/S Remotas Profibus-PA** foi extremamente testado e inspecionado antes de ser entregue ao usuário. Entretanto, durante o seu desenvolvimento, foi dada a possibilidade de reparos pelo usuário, quando necessário.
Em geral, é recomendado que não se repare as placas eletrônicas. Ao invés disso, o usuário deve ter partes sobressalentes, as quais podem ser adquiridas com a SMAR quando necessárias.

DIAGNÓSTICOS E SOLUÇÕES	
SINTOMA	**PROVÁVEL FONTE DO ERRO**
SEM CORRENTE QUIESCENTE	*Conexão de E/S do **DC303**:* *Verifique conexões segundo a polaridade e continuidade.* *Fonte de Alimentação:* *Verifique o sinal de saída da fonte de alimentação. A tensão de alimentação na borneira do **DC303** deve estar entre 9 e 32 Vdc.* *Falha no circuito eletrônico:* *Verifique as placas eletrônicas, trocando-as pelas sobressalentes.*
SEM COMUNICAÇÃO	Conexões com a rede de trabalho: Verifique as conexões com a rede, como os equipamentos, fontes de alimentação e terminadores. Impedância da rede: Verifique a impedância da rede (da fonte de alimentação e dos terminadores). Configuração do Mestre: Verifique a configuração de comunicação e parametrização do mestre. Configuração da rede: Verifique a configuração de comunicação na rede de trabalho. *Falha no circuito eletrônico:* *Verifique as placas eletrônicas, trocando-as pelas sobressalentes.*
ENTRADAS INCORRETAS	Conexão dos terminais de entrada: Verifique a polaridade e continuidade. Fonte de alimentação das entradas: *Verifique a alimentação. A tensão deve estar entre 18 e 30 Vdc e o consumo típico quando todas as entradas estão ativas é 120mA.*
SAÍDAS INCORRETAS	Conexão dos terminais de saída: Verifique polaridade e continuidade. Fonte de alimentação das saídas: *Verifique a alimentação. A tensão deve estar entre 20 e 30 Vdc e a máxima corrente de saída é de 0.5 A.*

Procedimento de desmontagem

Refira-se a Figura 4.1 - Vista Explodida do **DC303**. Certifique-se de que tenha desconectado a fonte de alimentação antes de desmontar o **DC303**.

ATENÇÃO
As placas possuem componentes CMOS que podem ser danificados por descargas eletrostáticas. Observe os procedimentos corretos para manipular estes componentes. Também é recomendado armazenar as placas de circuito em embalagens à prova de descargas eletrostáticas.

Solte as travas laterais que prendem a caixa principal do invólucro e, então, a trava principal. A placa principal e a de E/S poderão, então, ser acessadas.
Para removê-las, retire os parafusos que fixam-nas e manuseie cuidadosamente, sem danificá-las.

Procedimento de Montagem

- Coloque as placas cuidadosamente em suas posições no invólucro.
- Aperte os parafusos de fixação das mesmas.
- Certifique-se de que as conexões entre as mesmas estejam corretas.
- Observe a posição dos LEDs e cuidadosamente encaixe a tampa principal, travando-a lateralmente e depois através da trava principal.

Procedimentos de Atualização do Firmware do DC303

Para atualizar o firmware do **DC303** refira-se ao manual da **FDI302**, disponível no website da Smar: www.smar.com.br .

Acessórios

A placa principal e a de E/S podem ser trocadas independentemente.

ACESSÓRIOS	
CÓDIGO DE PEDIDO	**DESCRIÇÃO**
PBI-PLUS	Interface USB para PROFIBUS-PA.
SYSCON	Ferramenta de Configuração.
PS302	Fonte de Alimentação.
BT302	Terminador.
FDI-302-2	Interface de download de firmware para dispositivo de campo PROFIBUS-PA
DF47	Barreira de Segurança Intrínseca.

Vista Explodida

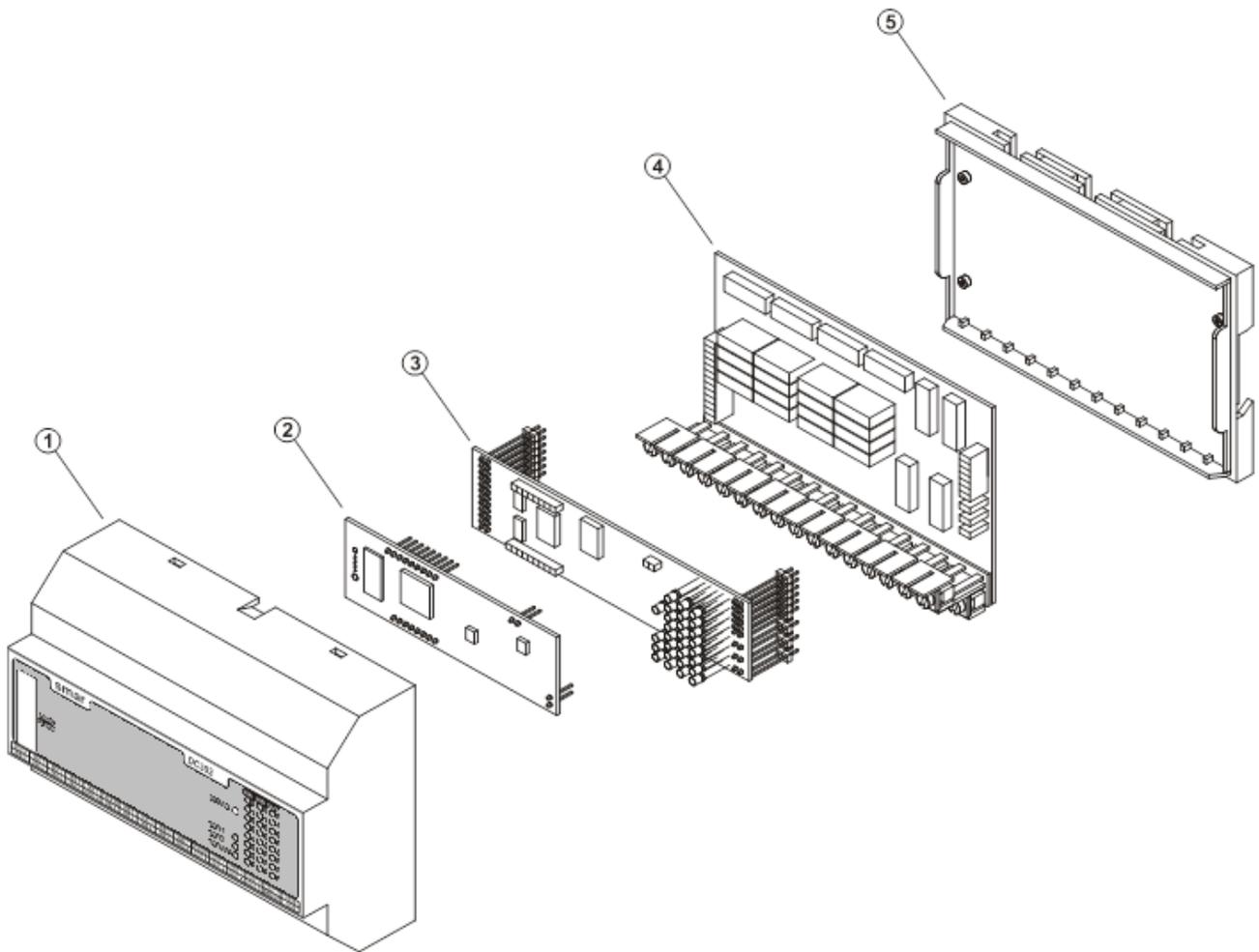



Figure 4.1 – Vista Explodida do DC303

Relação de Peças Sobressalentes

RELAÇÃO DE PEÇAS SOBRESSALENTES		
NOME	POSIÇÃO	CÓDIGO
Invólucro	1 e 5	400-1224
Placa Eletrônica Principal	2	400-1221
Placa de Interface	3	400-1222
Placa de E/S	4	400-1223

CARACTERÍSTICAS TÉCNICAS

Geral

Sinal (Comunicação)	Digital. Modo tensão Profibus-PA 31,25 Kbits/s, de acordo com a IEC 61158-2.
Fontes de Alimentação	Onde se requer isolação entre as fontes das entradas e saídas, recomenda-se usar no mínimo duas fontes de alimentação, sendo uma para as entradas e outra para as saídas e Vdc. Onde a aplicação não exigir isolação entre as fontes, pode-se ter somente uma fonte para entradas, saídas e Vdc. As saídas e entradas são isoladas opticamente.
Consumo de corrente quiescente	150 mA da fonte Vdc.
Corrente de Partida	400 mA durante os primeiros 20 s após a partida.
Tempo de partida	Aproximadamente 10 segundos.
Tempo de atualização	Aproximadamente 60 ms. O tempo de atualização está relacionado com a atualização das entradas e saídas do **DC303**, não depende do macrocycle do sistema.
Impedância de saída	Não Intrinsecamente seguro de 7,8 kHz - 39 kHz deve ser maior ou igual a 3 kΩ. Intrinsecamente seguro: (assumindo barreira de SI na alimentação) de 7,8 kHz – 39 kHz maior ou igual a 400 Ω.
Blocos Funcionais	Até 16 DIs (Discrete Input Block) e até 8 DOs (Discrete Output Block) e ainda possui um FFB, Bloco Flexivel (Built-in) que permite a execução de lógicas tais como: AND, OR, XOR e NOT. Funções como: Timer On-Delay, Timer Off-Delay, Timer Pulse, Pulse Counter Down (CTD), Pulse Counter Up (CTU), Flip-Flop RS e Flip-Flop SR.
Efeito de Vibração	De acordo com SAMA PMC 31.1.
Limites de Temperatura	**Operação:** -40 a 85ºC (-40 a 185ºF). **Armazenagem:** -40 a 110ºC (-40 a 230ºF).
Invólucro	**Invólucro/base:** Policarbonato, 10% fibra de vidro. **Terminais:** Parafusos em Zinco, aço cromado. **Faixa de Temperatura:**110ºC (230 ºF) UL94VO. **Proteção:** IP20 (toque com o dedo) e VBG4 e outros requisitos europeus de prevenção de acidentes. Pode ser opcionalmente fornecido em caixa de distribuição a prova de explosão para montagem no campo.
Configuração	Via comunicação Profibus usando ferramentas baseadas em EDDL ou FDT/DTM.
Montagem	Usando trilho DIN (TS35-DIN EN 50022 ou TS32-DIN EN50035 ou TS15-DIN EN50045).

Entradas do DC303

Descrição das Entradas

As entradas recebem tensões DC e convertem em sinal lógico Ligado ou Desligado.
Possui um grupo de 16 entradas isoladas oticamente que recebem 24 Vdc.

Na falha da fonte de alimentação das entradas haverá indicação via diagnósticos cíclicos.

Especificações Técnicas

Arquitetura	O número de entradas é 16.
Isolação, os grupos são individualmente isolados	Isolação Ótica de 5000 Vac.
Fonte Externa	18 - 30 Vdc.
Consumo Típico do grupo	120 mA (todas as entradas em estado ligado).
Indicador de Alimentação	LED verde.

Entradas	Nível no estado LIGADO (Verdadeiro Lógico) 15 - 30 Vdc.
	Nível no estado DESLIGADO (Falso Lógico) 0 - 5 Vdc.
Impedância Típica	3,9 kΩ.
Indicador do Estado de ativação	LED vermelho.
Informação de Chaveamento	Tempo de "0" a "1": 30 μs.
	Tempo de "0" a "1": 50 μs.
Cabeamento	Fio único: 14 AWG (2 mm²) 14 AWG (2 mm²).
	Dois fios: 20 AWG (0,5 mm²).

DC303: Saídas em Coletor Aberto

Descrição - Saídas

As saídas estão projetadas com transistores na configuração NPN, coletor aberto de forma a trabalhar com relés, solenóides e outras cargas DC com 0,5 A por saída. Todo o grupo de saídas compartilha o mesmo terra e elas são isoladas uma das outras e da rede Profibus-PA.

Na falha da fonte de alimentação das saidas haverá indicação via diagnósticos cíclicos.

Especificações Técnicas

Arquitetura	Número de saídas: 8.
Isolação	Isolação Ótica de 5000 Vac.
Fonte Externa	20 a 30 Vdc.
Consumo Máximo	35 mA.
Indicação de Alimentação	LED verde.
Saídas	Tensão máxima chaveada: 30 Vdc.
	Tensão Máxima de Saturação 0,55 V a 0,5 A.
	Máxima Corrente por saída: 0,5 A.
	Indicação de saída ativa: LED vermelho.
	Lógica da indicação: Ligado quando o transistor estiver ligado.
	Máxima Corrente de "Leakage": 100 μA a 35 Vdc.
Condição das Saídas durante: **Partida (Power-Up)** **Atualização do Firmware**	DESLIGADO.
Proteção independente das saídas	Desligamento Térmico: 165°C.
	Histerese Térmica 15°C.
	Proteção de sobre-corrente: 1,3 A a 25 Vdc máximo.
Diodo Clamp, informação de chaveamento	Tempo de 0 a 1: 250 μs.
	Tempo de 1 a 0: 3 μs.
Cabeamento	Fio único: 14 AWG (2 mm^2).
	Dois fios: 20 AWG (0,5 mm^2).

Código de Pedido

MODELO	DESCRIÇÃO
DC303 - E/S Remotas Profibus-PA	1 grupo de 16 entradas a 24 Vdc isoladas oticamente. 1 grupo de 8 saídas coletoras abertas isoladas oticamente.

Apêndice A

smar

FSR - Formulário para Solicitação de Revisão

Entrada e Saída Remota Profibus-PA

DADOS GERAIS

Modelo: DC303

Nº de Série: ____________________

TAG: ____________________

Utilizando quantos canais?
ENT 1-4 ( ) 9-12 ( )
5-8 ( ) 13-16 ( )
SAI 1-4 ( ) 5-8 ( )

Configuração: PC ( ) Software: ____________ Versão: ____________

DADOS DA INSTALAÇÂO

Tipo/Modelo/Fabricante do equipamento conectado ao DC303: ____________________

____

DADOS DO PROCESSO

Classificação da Área/Risco: ( ) Sim, por favor especifique: ____________________
( ) Não
Mais detalhes: ____________________

Tipos de Interferência presente na área: Sem interferência ( ) Temperatura ( ) Vibração ( ) Outras: ____________

Temperatura Ambiente: De __________ºC até __________ºC.

DESCRIÇÃO DA OCORRÊNCIA

____

____

____

____

SUGESTÃO DE SERVIÇO

Ajuste ( ) Limpeza ( ) Manutenção Preventiva ( ) Atualização / Up-grade ( )

Outro: ____________________

DADOS DO EMITENTE

Empresa: ____________________

Contato: ____________________

Identificação: ____________________

Setor: ____________________

Telefone: _______ ____________ _______ ____________ **Ramal:** ____________

E-mail: ____________________ **Data:** ______/ ______/ ______

Verifique os dados para emissão da Nota Fiscal de Retorno no Termo de Garantia disponível em: http://www.smar.com/brasil/suporte.asp

Retorno de Materiais

Caso seja necessário retornar o **DC303** para avaliação técnica ou manutenção, basta contatar a empresa SRS Comércio e Revisão de Equipamentos Eletrônicos Ltda., <u>autorizada exclusiva da Smar</u>, informando o número de série do equipamento com defeito, enviando-o para a SRS de acordo com o endereço contido no termo de garantia.

Para maior facilidade na análise e solução do problema, o material enviado deve conter, em anexo, a documentação descrevendo detalhes sobre a falha observada no campo e as circunstâncias que a provocaram. Outros dados, como local de instalação, tipo de medida efetuada e condições do processo são importantes para uma avaliação mais rápida e para isto, use o Formulário para Solicitação de Revisão (FSR).